Security Testing Report

EDI\CI

MEMO v1.3

EDI\CI

Table of Contents

Document Version Control

Name of the document :MEMO v1.3 Security Testing Report			
Version: 1.0	**Intake ID:**	2665	
Document Definition: This document highlights the vulnerabilities currently existing in the application under scope. It also documents possible actions to be taken to reduce/eliminate the vulnerabilities.	**Document ID:**	PRHC/C40/SVN/2944	
Author: Sai Praneetha Bhaskaruni, Harshal Kukade, Navin Kumar Pari	**Effective Date:**	**09/Aug/2023**	
Reviewed by: Karthik Lalan			

Document History

Version	Date	Author	Section	Changes
1.0	**09/Aug/2023**	**Sai Praneetha Bhaskaruni, Harshal V, Navin Kumar Pari**	**Complete**	**Assessment as part of Intake 2665**

Distribution List

User/Department/Stakeholder	E-Mail ID	
Project Owner and PSO	Chavali, Kesavanand	|kesavanand.chavali@philips.com
	Harikumaran	|harikumaran.j@philips.com
	Ramamoorthi, Sagar	| sagar.ramamoorthi@philips.com

	Lohiteshwarappa, Kishore	|kishore.hl@philips.com
	rohan.dcunha	|rohan.dcunha@philips.com
	SHINDHE, PREETHIKA	|preethika.shindhe@philips.com

1. Definitions & Abbreviations

Term	Explanation
SCoE	Security Center of Excellence
TLS	Transport Layer Security
SSL	Secure Socket Layer
XSS	Cross Site Scripting
CORS	Cross Origin Resource Sharing

The severity of every vulnerability has been calculated by using industry standard **Common Vulnerability Scoring System (CVSS)** used for assessing the severity of computer system vulnerabilities. CVSS provides a way to capture the principal characteristics of a vulnerability and produce a numerical score (Scores range from 0 to 10, with 10 being the most severe) reflecting its severity. The numerical score can then be translated into a qualitative representation (such as low, medium, high, and critical) to help organization properly assess and prioritize their vulnerability management processes.

The severity rating for the numerical values are mapped below:

None	0.0
Low	0.1 – 3.9
Medium	4.0 – 6.9
High	7.0 – 8.9
Critical	9.0 – 10.0

The **Severity** and **CVSS vector** of each vulnerability is calculated using the CVSS V3 **Base Score Metrics** Calculator located here. Vulnerabilities identified during security assessment are classified into standardized categories. Refer following table for more information:

Categories for vulnerability classification

Web application security assessment	OWASP Top Ten – 2021
Mobile application security assessment	OWASP Top Ten – 2016
IoT/Hardware security assessment	OWASP Top Ten – 2014

2. System Details & Architecture

The brief about the product architecture is explained below:

MEMO is a cloud-based Monitoring, Alerting and Analytics solution. Prometheus is at the heart of the solution, and it scrapes logs from product exporter and windows exporter components. Collected logs are uploaded to Thanos for further processing.

Testing environment: PenTest

Architecture Diagram: NA

3. Scope

The scope of this security assessment is to perform ***Grey-Box*** security testing to find security threats that may come from a malicious outsider or insider user of the **MEMO 1.3.** Security testing on ***Web Application, Web Services and Thick Client Application*** of the MEMO is performed.

The following list includes few examples of major activities performed during the assessment:

Web Application/ Web Services:

- Crawl through complete scope of the web application/service space and identify for any unauthenticated URL or directory.
- Check for all input injection based attacks across all the possible entry fields in Web API.
- Exploiting any known component vulnerability or service misconfiguration.
- Reviewing the transport layer security implemented.
- End-To-End Testing for Octopus WebApp.
- End-To-End Testing for 3WebServices.

Thick Client Application:

- End-To-End Full Application Testing

Follow "Test case execution" section for to get the detailed about test cases.

The test scope for this release is explained in the below table:

Start Date	End Date	Applications/Devices/IP's/URL's
15/JUN/2023	**03/JUL/2023**	**WebApp:** 1. Grafana URL: https://mas-edi-grafana-client-test.eu-west.monitoring.hsdp.io/login 2. Octopus Remote Deploy: https://philips-ei.octopus.app/app#/ • Version: v1.3 • Environment: Test • User Role: ServiceUser1
		WebAPI/ web services: 1. MEMO-ITaap: https://dev.apps.api.it.philips.com/test/solutionsit/memo-event-mgmt/event 2. All sticky notes API's: https://philipsinformatics-alertsink-dev.eu1.phsdp.com/api/StickyNote

		3. MEMO backend API Collection: https://philipsinformatics-alertsink-test.eu1.phsdp.com/api/CRM/Case • Version: v1.3 • Environment: Test • Credentials Available
11/JUL/2023	**08/Aug/2023**	• **Thick Client Applications:** 1. Philips.MEMO.ProductExporter.msi 2. Philips.MEMO.Prometheus.Win.msi 3. Philips.MEMO.snmp.Win.msi 4. Philips.MEMO.WindowsExporter.msi 5. OctopusTentacle.latest-x64.msi ○ Version: v1.3 ○ Environment: Test ○ Credentials Available: memoconc\fse_Admin

4. Out of Scope

Below mentioned items are out of scope for the current security assessment:

- Source code review
- Stress test (DDOS)

5. Executive Summary

Security Center of Excellence (ScoE) team is engaged in activities to conduct security assessment of **MEMO – 1.3** which included **Web Application, Web Services and Thick Client App Security Testing** in scope. The purpose of the engagement is to evaluate the security of the **MEMO – 1.3**

Note: Only highlights of important vulnerabilities are described below. Please refer 'Vulnerability Summary' section for complete detailed list of vulnerabilities.

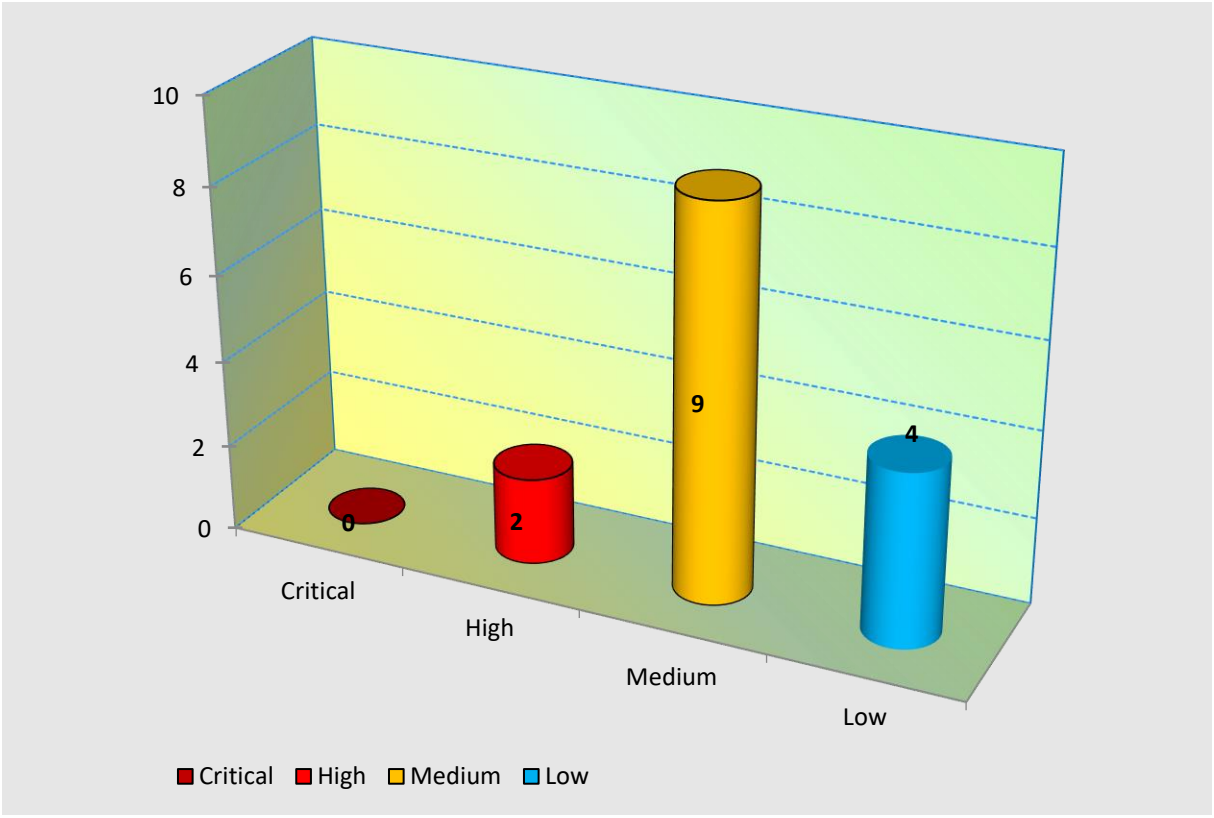
During the security assessment **following factors are found with consideration for significant improvement**:

- Authentication Bypass
- Information disclosure via internal API misconfiguration
- Broken Authentication via exposed internal API key
- API Supports Basic Authentication
- CORS (Cross Origin Resource Sharing) Misconfiguration
- Lack of Binary Protection & Code Signing
- Information Disclosure
- Insecure HTTP Methods Allowed
- TLS Implementation Flaws
- Weak Password Complexity Requirements
- Improper Session Management
- Improper Error Handling
- Lack of Input Validation
- Extraneous Services Enabled
- Weak Input validation and Data Not Validated for Semantic Correctness

VULNERABILITY SUMMARY CHART

The graph below shows a summary of the number of vulnerabilities and their severities.

Note: The vulnerabilities mentioned in this report are technical vulnerabilities only. The Product Security Risk Assessment would report the business risks associated with these vulnerabilities.



6. Vulnerability Summary

The findings and vulnerabilities from the assessment are explained in the below table:

Finding No.	Vulnerability Title	Technical Risk	Impacted Area	CVE ID*	Status as of on 30Jan2023	Status as of on 08Aug2023
1	Authentication Bypass	High	API	NA	NA	Open
2	Information disclosure via internal API misconfiguration	Medium	API	NA	NA	Open
3	Broken Authentication via exposed internal API key	High	API	NA	NA	Open
4	API Supports Basic Authentication	Medium	API	NA	NA	Open
5	CORS (Cross Origin Resource Sharing) Misconfiguration	Medium	API	NA	NA	Open
6	Lack of Binary Protection & Code Signing	Medium	DesktopApp	NA	NA	Open
7	Information Disclosure	Medium	DesktopApp	NA	NA	Open
8	Insecure HTTP Methods Allowed	Medium	DesktopApp	NA	NA	Open

9	TLS Implementation Flaws	Medium	DesktopApp	NA	NA	Open
10	Weak Password Complexity Requirements	Medium	WebApp	NA	NA	Open
11	Improper Session Management	Medium	WebApp	NA	NA	Open
12	Improper Error Handling	Low	Desktopapp	NA	NA	Open
13	Lack of Input Validation	Low	API & DesktopApp & Webapp	NA	NA	Open
14	Extraneous Services Enabled	Low	Desktopapp	NA	NA	Open
83667	HTTPS Fallback Allowed	Low	Webservices	NA	Open	Closed
83668	Weak Input validation and Data Not Validated for Semantic Correctness	Low	WebApp & WebServices	NA	Open	Open
83669	Verbose Error Messages	Low	Webservices	NA	Open	Closed

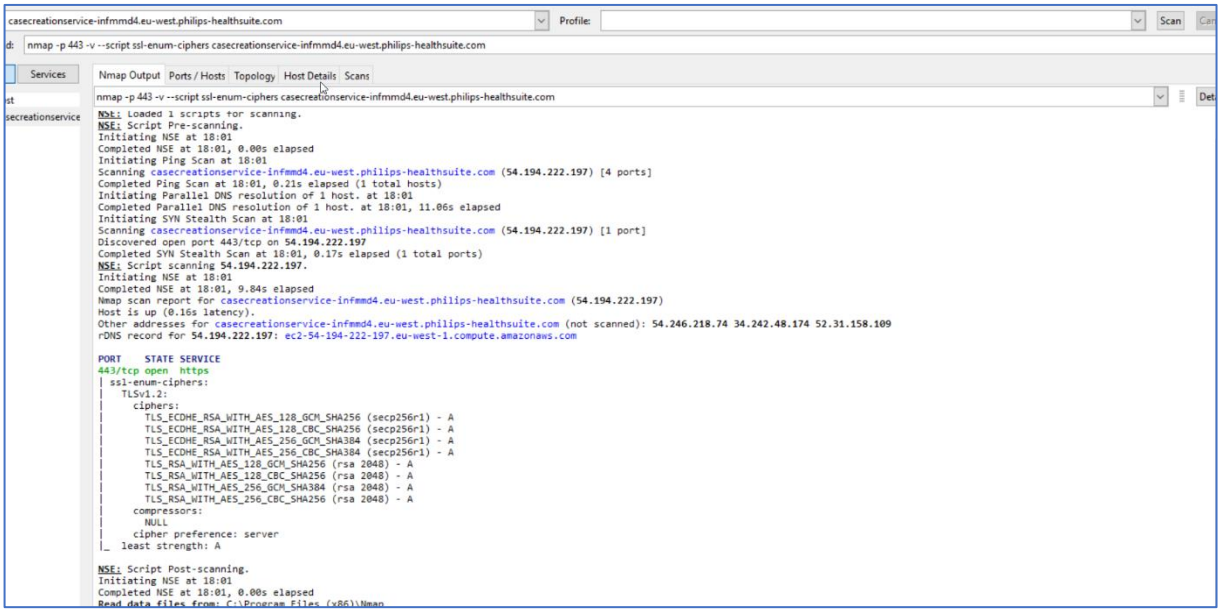
*CVE ID are mentioned for the vulnerabilities which has a known external CVE.

7. Observations

Below mentioned observations are not considered as vulnerability but informative to the business.

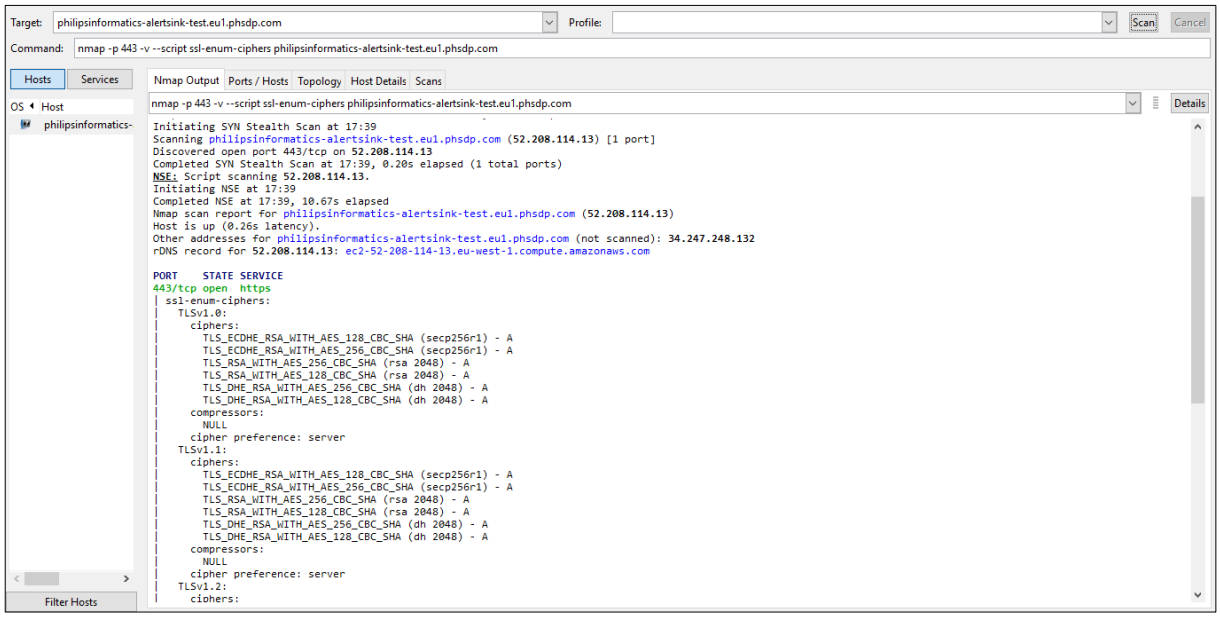
Observations which shows good implementation or **best practice** identified

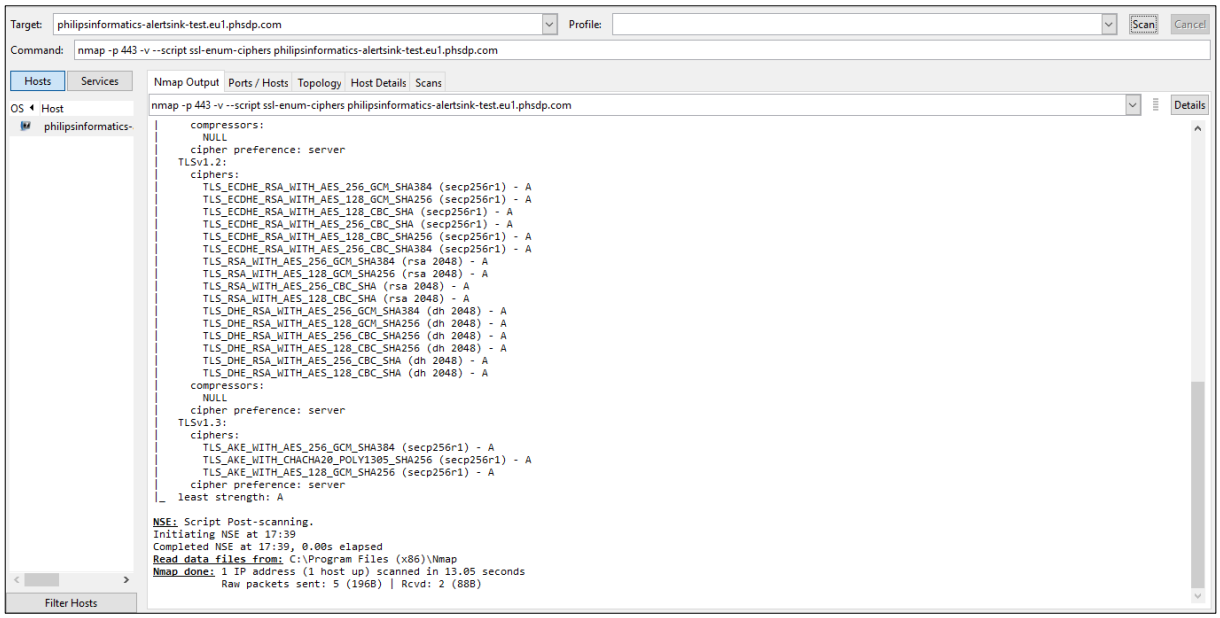
- The usage of the required HTTP methods has been defined properly at the server end. Which means if a GET request is used and if a different request is tried, then the server responds with “405” method not allowed error which is a good observation.
- The server of create case API is found to be having A grade cipher suites



Retest as of on 15June2023:

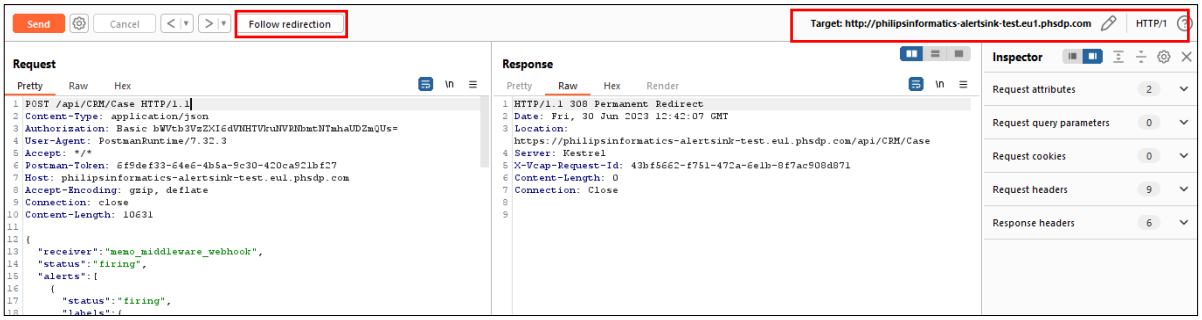
- TLS1.2 supported with weak CBC cipher, which is not a best practice.

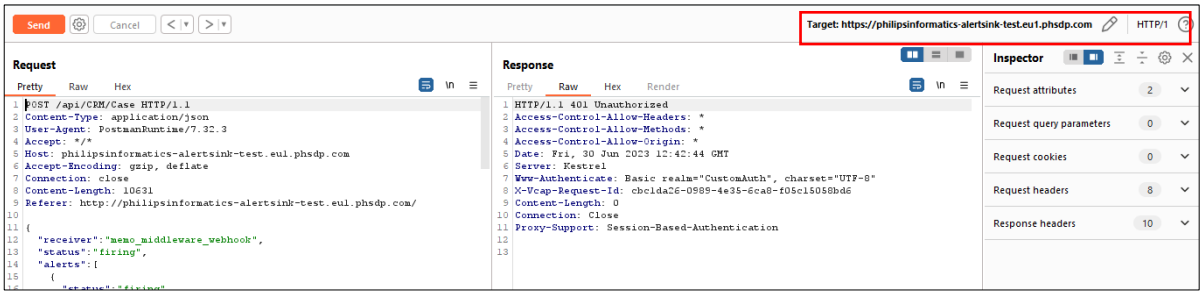




Weak TLS cipher

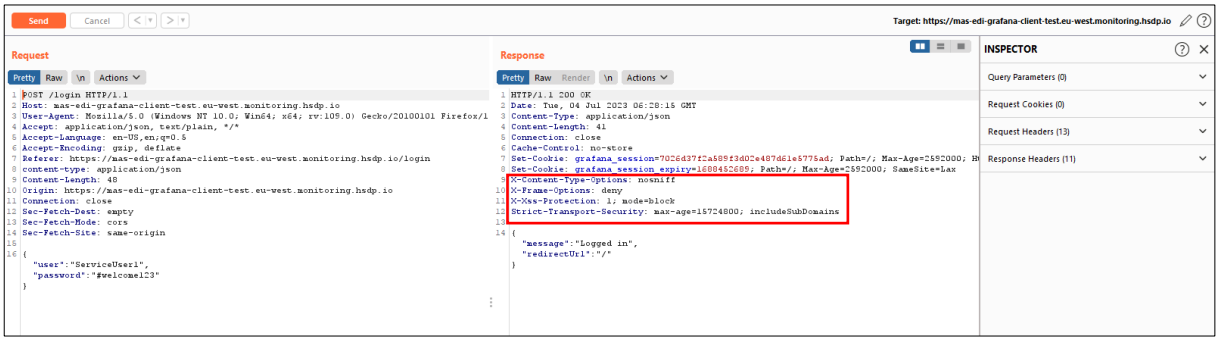
- It is observed that the application is not accessible over HTTP. When we tried to send the request over HTTP, the application is redirecting to HTTPS.





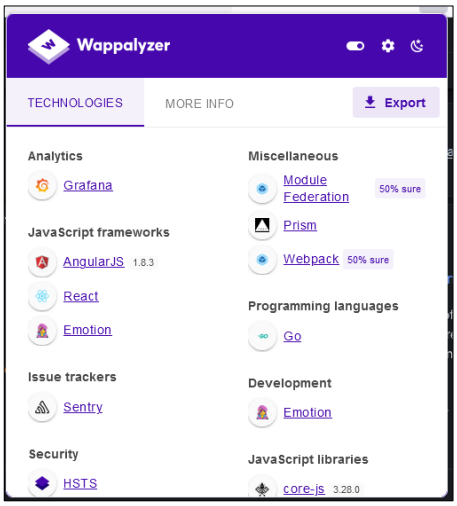
Retest as of on 15June2023:

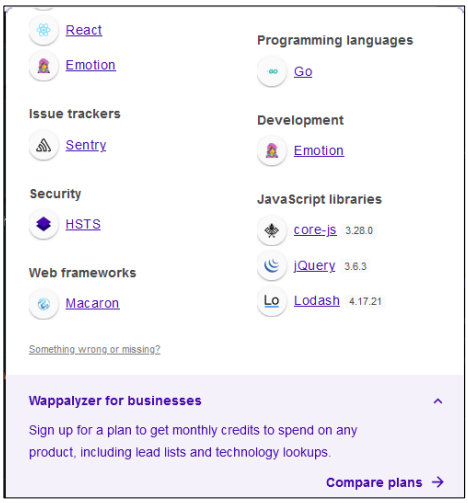
- HTTP Security Headers implemented: -
  1. HSTS (HTTP Strict Transport security) Strict-Transport-Security: max-age includesubdomain “preload”
  2. X-Content-Type-Options: nosniff (Browser MIME type sniffing is disabled)
  3. X-Frame-Options: DENY (The page cannot be displayed in a frame, regardless of the site attempting to do so)
  4. X-Xss-Protection: 1

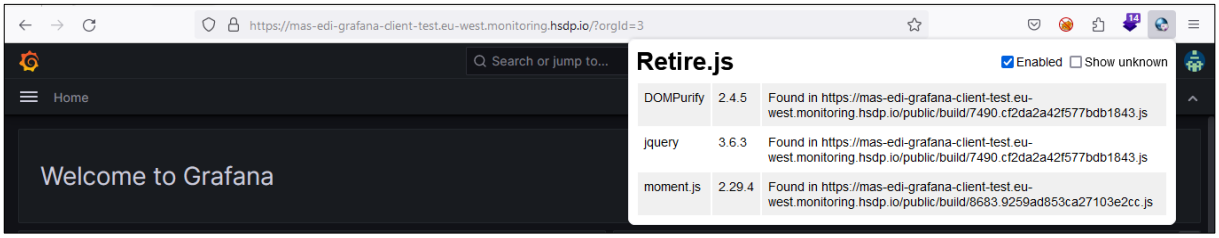


- Technologies found to be vulnerable.

Technologies	Current Version	Latest Available Version	Remarks	Reference
AngularJS	1.8.3	1.8.3	No vulnerabilities identified for current package	https://security.snyk.io/package/npm/angular/1.8.3
Core-js	3.28.0	3.31.0	No vulnerabilities identified for current package	https://security.snyk.io/package/npm/core-js
jQuery	3.6.3	3.7.0	No vulnerabilities identified for current package	https://security.snyk.io/package/npm/jquery/3.6.3
Lodash	4.17.21	4.17.21	No vulnerabilities identified for current package	https://security.snyk.io/package/npm/lodash/4.17.21
DOMPurify	2.4.5	3.0.4	No vulnerabilities identified for current package	https://security.snyk.io/package/npm/dompurify
moment.js	2.29.4	2.29.4	No vulnerabilities identified for current package	https://security.snyk.io/package/npm/moment/2.29.4
React	NA	NA	NA	NA
Emotion	NA	NA	NA	NA
HSTS	NA	NA	NA	NA
Go	NA	NA	NA	NA



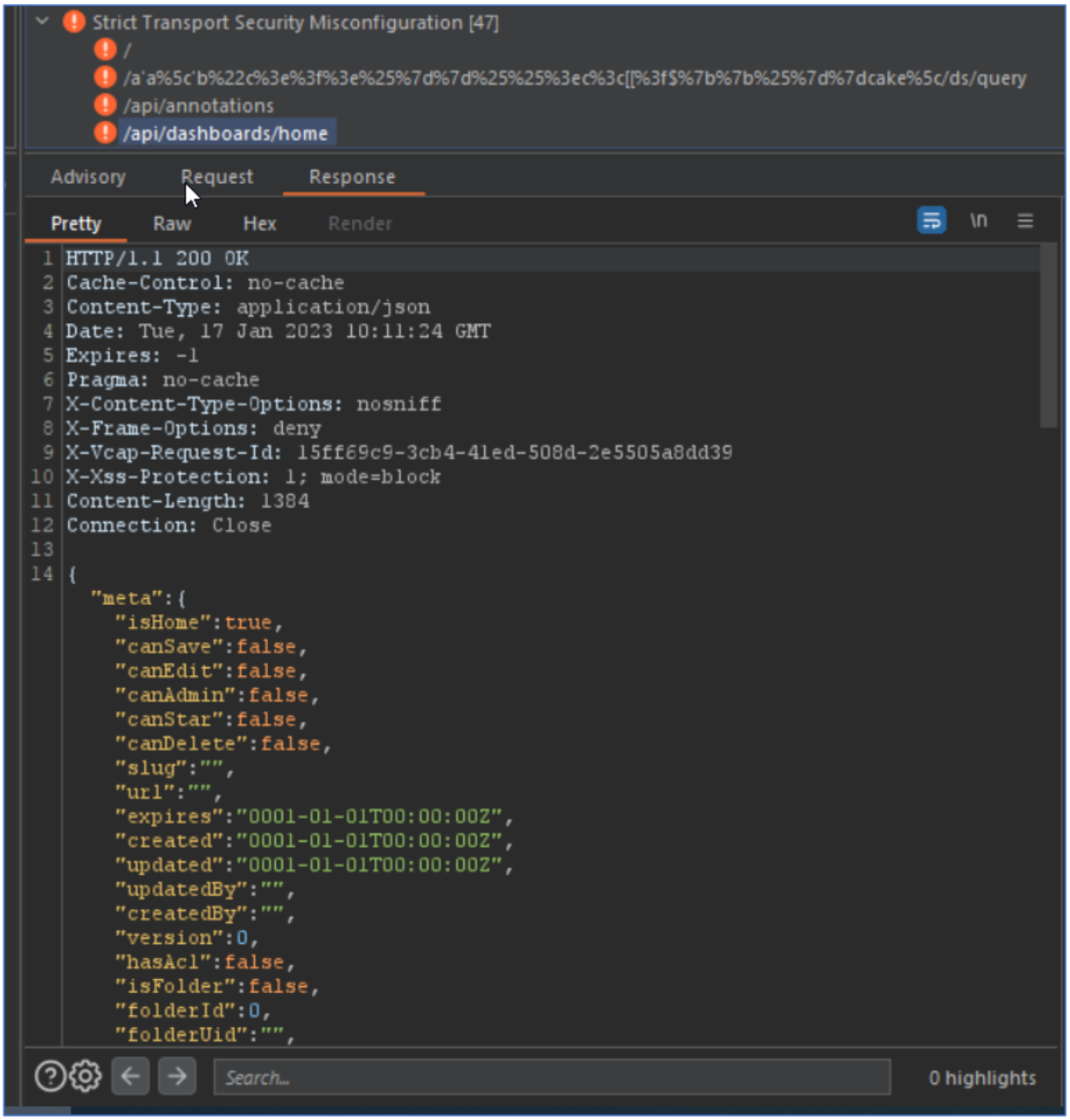




Observations which show missing best practice or possible weak implementation (this may/may not be direct active threat):

- It is observed that the application uses HTTP/1.1 protocols which is not as robust as HTTP/2.0
- Create Case web API shared by the business team is accessible via open internet which is not a best practice.

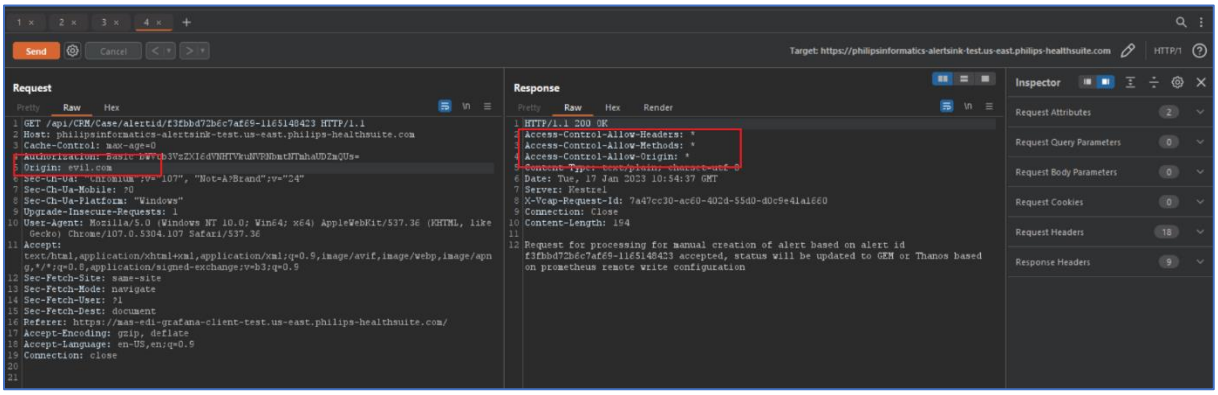
- HTTP Strict Transport Security is not properly implemented in the Grafana application



Retest as of on 15June2023:

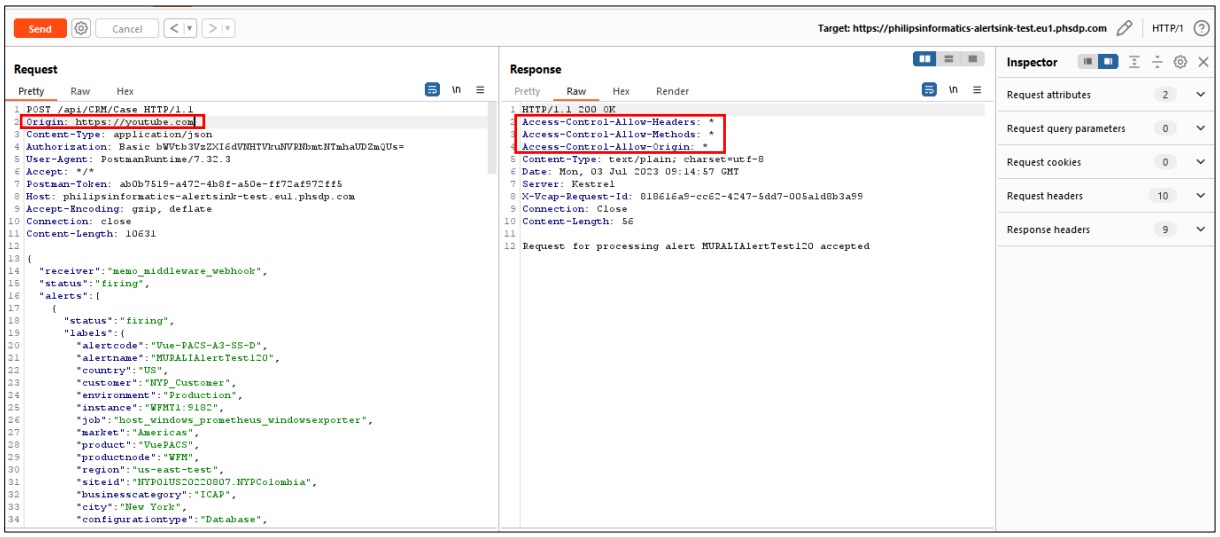
- HTTP Strict Transport Security is properly implemented in the Grafana application.

- It is observed that the case creation endpoint has set with wildcard for access control allow origin header which is a cors misconfiguration.

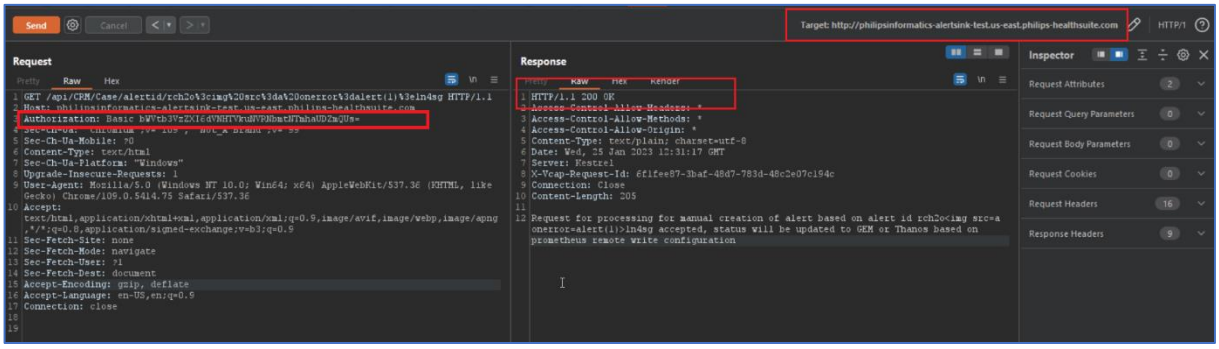


Retest as of on 15June2023:

- It is observed that the case creation endpoint has set with wildcard for access control allow origin header which is a cors misconfiguration.

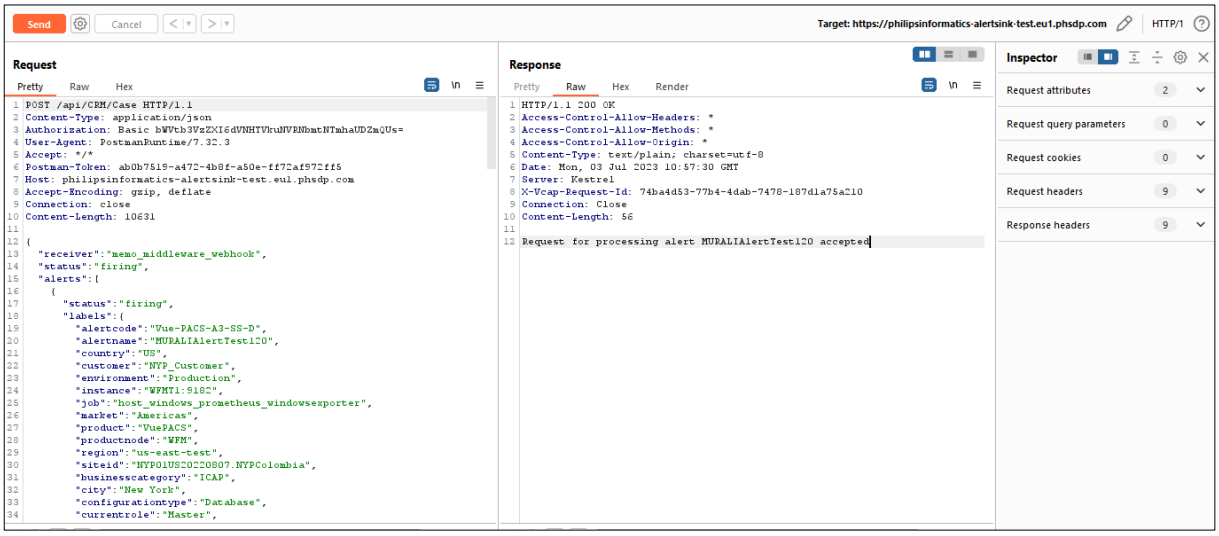


- It is observed that basic authentication is used for creating case in Grafana application which is not a best practice.



Retest as of on 15June2023:

- It is observed that basic authentication is used and the authentication mechanism was not working properly for creating case in Grafana application which is not a best practice.



Note: It is observed that the grafana web application allows user to directly query into the database. In the test environment, we found the web application non admin user can interact with the database and even modify/delete data. In response to this, the business team has updated ScoE that in production the user will have only read access in the database and write access will be disabled.

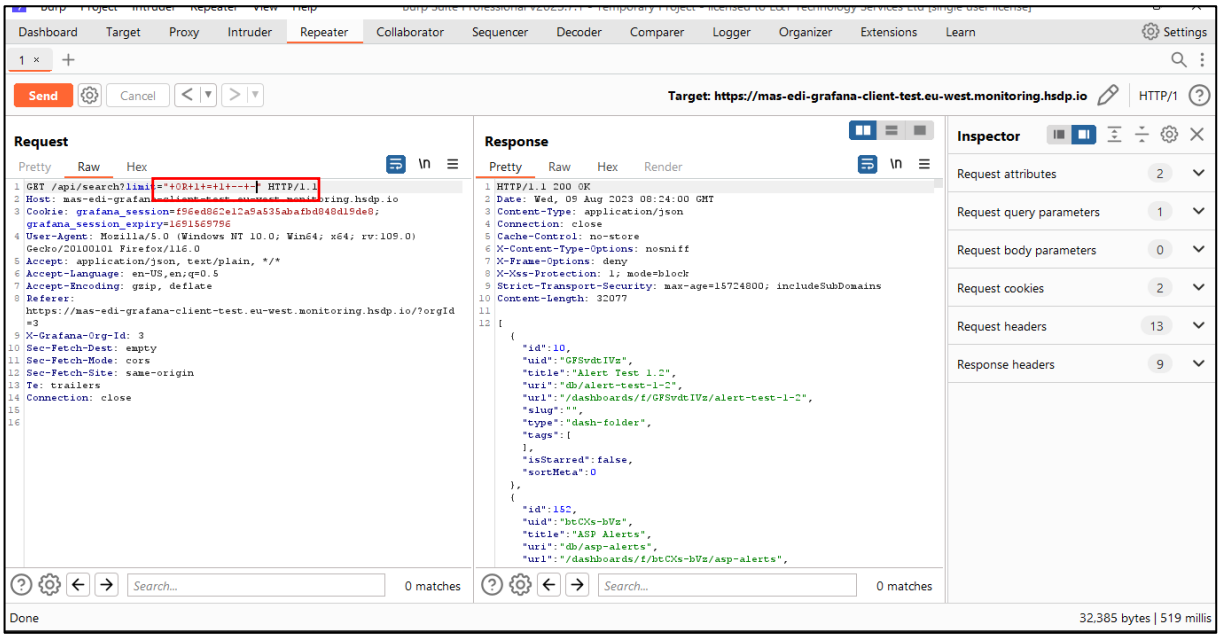
- It is observed that the application uses HTTP/1.1 protocols which is not as robust as HTTP/2.0.
- After attempting to log in with an incorrect password for 5 times, it is observed that the application does not return ”Your account is locked out”. Thereby confirming that the account is not locking out after 5 incorrect authentication attempts.
- It is observed that the application doesn’t set Content Security Policy (CSP) headers properly.
- While testing for XSS, it is observed that it is allowing PUT header which is not a best practice.

```
PUT /api/user HTTP/1.1
Host: mas-edi-grafana-client-test.eu-west.monitoring.hsdp.io
Cookie: grafana_session=c10be5cb17bf8a054f302e96796e16a2;
grafana_session_expiry=1687241367
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64;
rv:109.0) Gecko/20100101 Firefox/114.0
Accept: application/json, text/plain, */*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer:
https://mas-edi-grafana-client-test.eu-west.monitoring.hsdp.io
/profile
Content-Type: application/json
X-Grafana-Org-Id: 3
Content-Length: 96
Origin:
https://mas-edi-grafana-client-test.eu-west.monitoring.hsdp.io
Sec-Fetch-Dest: empty
Sec-Fetch-Mode: cors
Sec-Fetch-Site: same-origin
Te: trailers
Connection: close

{
  "name":"<ScriPt>alert(document.cookie)</ScriPt>",
  "email":"ServiceUser1",
  "login":"ServiceUser1"
}
```

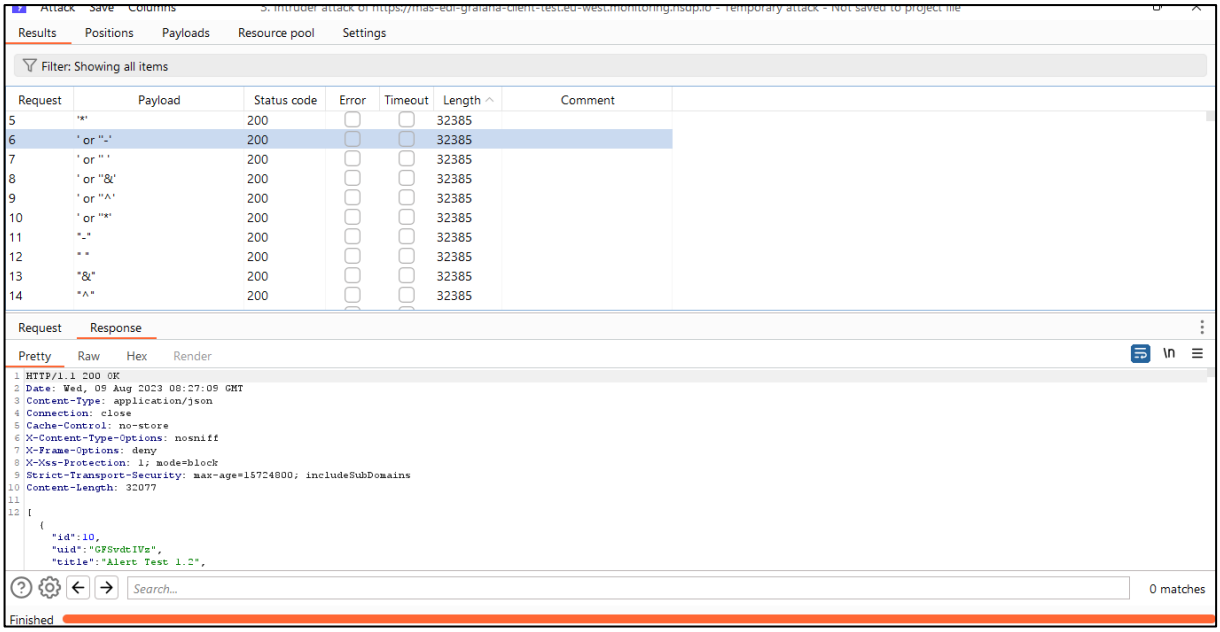
```
1  HTTP/1.1 200 OK
2  Date: Tue, 20 Jun 2023 06:09:03 GMT
3  Content-Type: application/json
4  Content-Length: 26
5  Connection: close
6  Cache-Control: no-store
7  X-Content-Type-Options: nosniff
8  X-Frame-Options: deny
9  X-Xss-Protection: 1; mode=block
10 Strict-Transport-Security: max-age=15724800; includeSubDo
11
12 {
     "message":"User updated"
   }
```

Retest as of on 15June2023:

- It is observed that input sanitization is not being done, which is a missing best practice for sql injection.



Invalid Input payload allowed.



Input validation not implemented. Getting 200 OK response for invalid entries.

8. Detailed Vulnerability Report

8.1 WebServices: Authentication Bypass

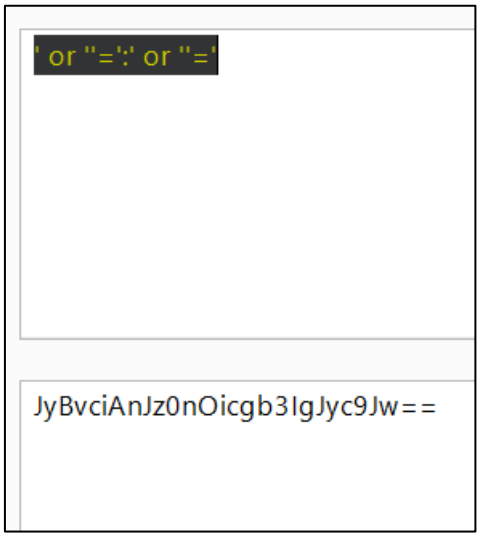
Vulnerability Title	API Authentication Bypass
Vulnerability Category	API2:2023:Broken Authentication
Severity	**High**
CVSS V3 Calculation	CVSS Base Score: 7.4 CVSS:3.0/AV:N/AC:H/PR:N/UI:N/S:U/C:H/I:H/A:N
Description	**Vulnerability Description:** It was observed that the authentication mechanism can be bypassed by using SQL queries in the username/password values. **Testing as of on 15Jun2023:** Payload used and it is base64 encoded chars. Authentication mechanism can be bypassed using SQL queries. **Exploitability rational:** Any user with access to this API endpoints will be able to carryout this attack by just passing SQL queries with base64 encoded. **Impact Rational:** A malicious user/attacker could use this vulnerability to gain unauthenticated access to the APIs and may be able to perform fraudulent actions, as well as gaining acess to sensitive information.
Affected Systems/IP Address/URL	https://philipsinformatics-alertsink-test.eu1.phsdp.com/api/CRM/Case **Testing as of on 15Jun2023:** https://philipsinformatics-alertsink-test.eu1.phsdp.com/api/StickyNote https://philipsinformatics-alertsink-dev.eu1.phsdp.com/api/CRM/Case

Recommendation	It is recommended to implement the password validation function properly. Allow login to the APIs only after verifying the username and password. It is recommended to implement proper server side and client-side validation of the password.
Status	**Open**

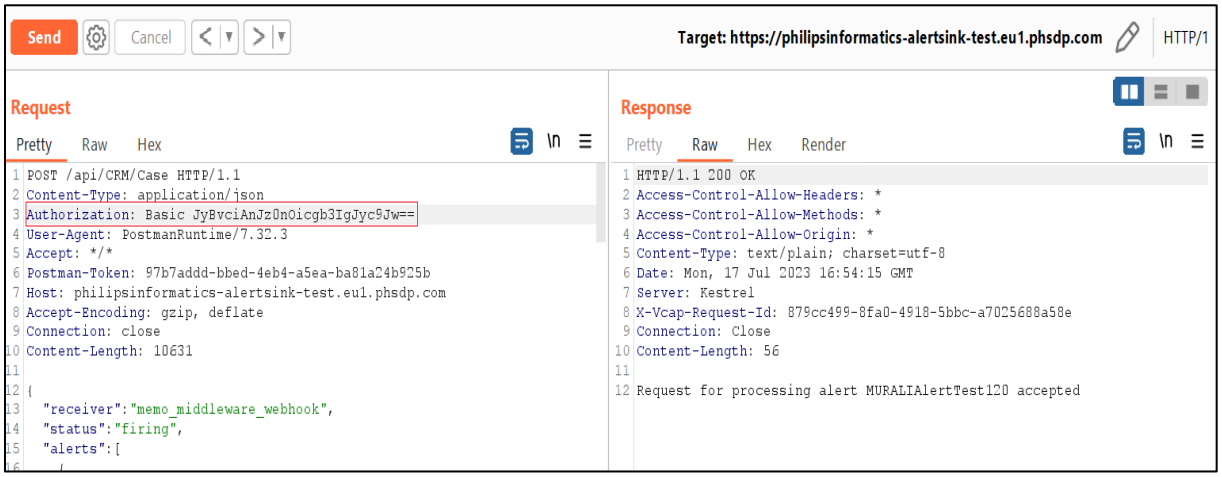
Steps to Reproduce:

- Load the APIs in the postman.
- Capture that request in the Burp suite and replace the authentication parameter with authentication bypass payload.
- You can observe that the payload output is successfully authenticate and got 200 responses.

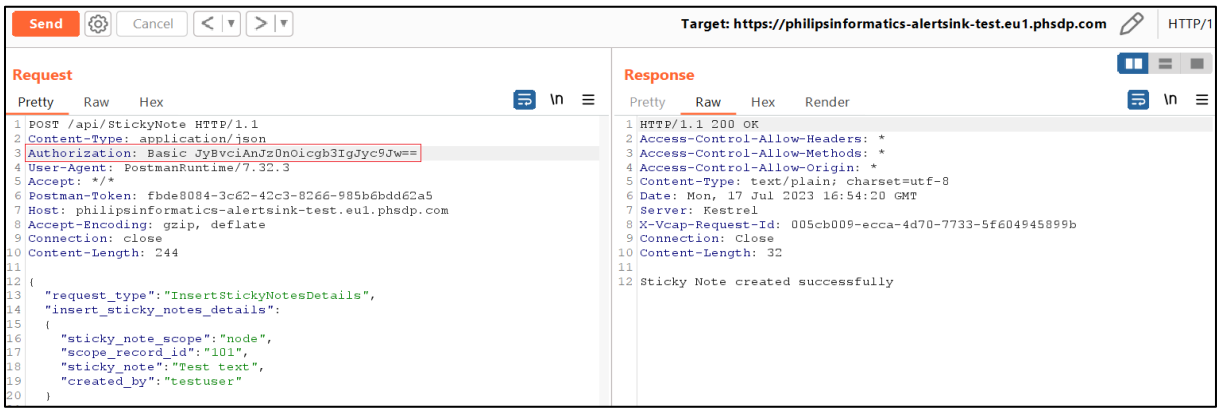
Supportive Evidence:



a. Screenshot shows that the payload used and it is base64 encoded chars.

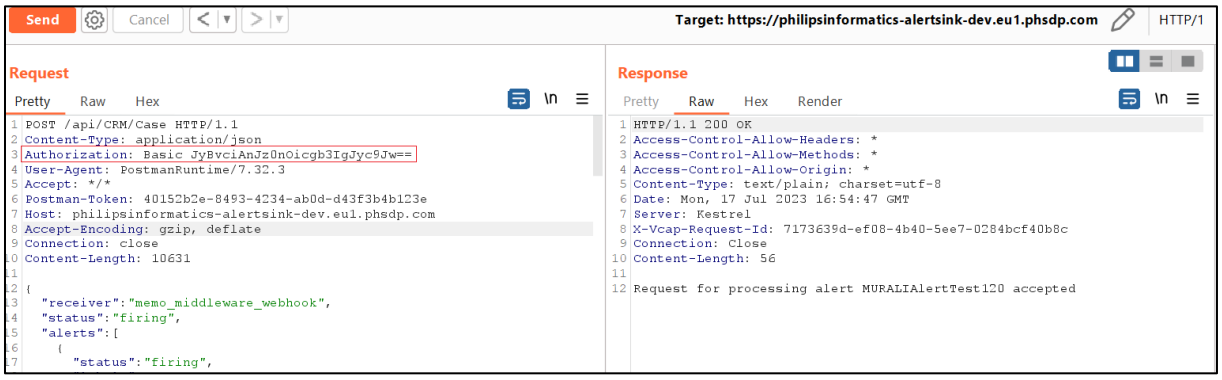


b. Screenshot shows that the CRM Case API accepting the payload passed in Authorization: Basic



c. Screenshot shows that the Sticky Note- Insert API accepting the payload passed in Authorization: Basic

d. Screenshot shows that the Sticky Note- Update API accepting the payload passed in Authorization: Basic



e. Screenshot shows that the CRM Case – dev API accepting the payload passed in Authorization: Basic

8.2 WebServices: Information disclosure via internal API misconfiguration

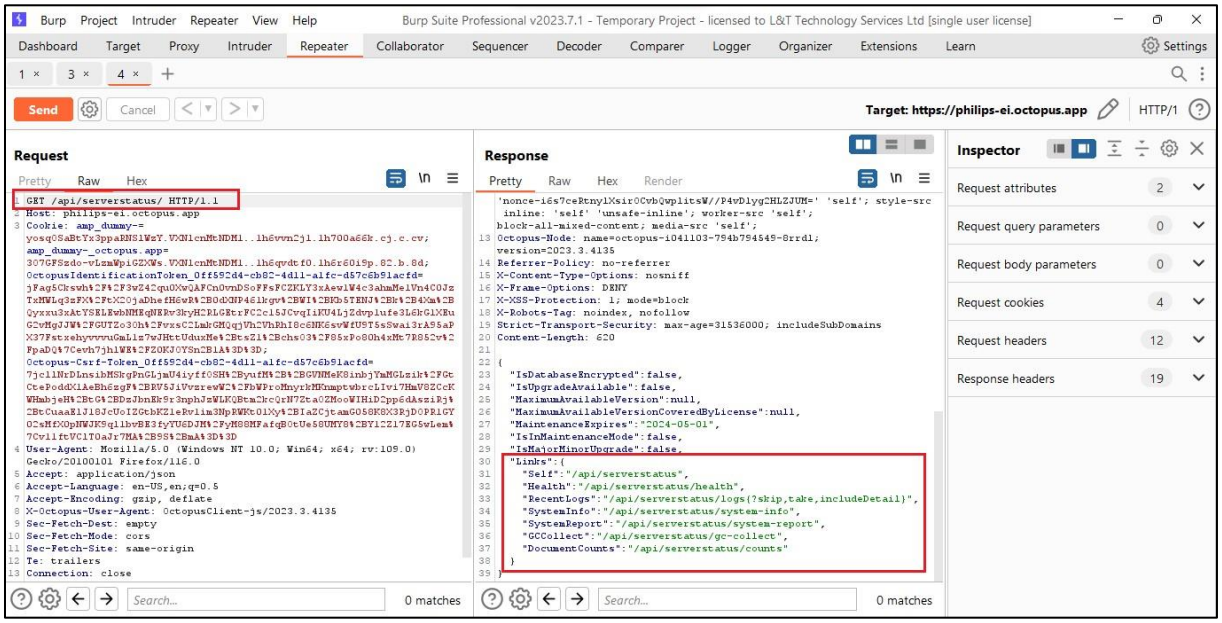
Vulnerability Title	Information disclosure via internal API misconfiguration
Vulnerability Category	A5- Security Misconfiguration
Severity	**Medium**
CVSS V3 Calculation	CVSS Base Score: 6.5 AV:N/AC:L/PR:L/UI:N/S:U/C:H/I:N/A:N
Description	**Vulnerability Description:** During the security assessment of the Web application, it is found that the application discloses the endpoints after adding / to end of the path. **Test as of on 26Jul2023:** Exposed internal API endpoints. **Exploitability Rational:** If an attacker accesses the application, he can able to retrieve each and every endpoints and he can traverse the path present on the application. **Impact Rational:** Attacker can able to understand the architecture and get more knowledge on the application.
Affected Systems/IP Address/URL	https://Philips-ei.octopus.app
Recommendation	Need to remove internal API endpoint from public user content. Leaking technical information such as directory structure can be prohibited.
Status	**Open**

Test as of on 26Jul2023:

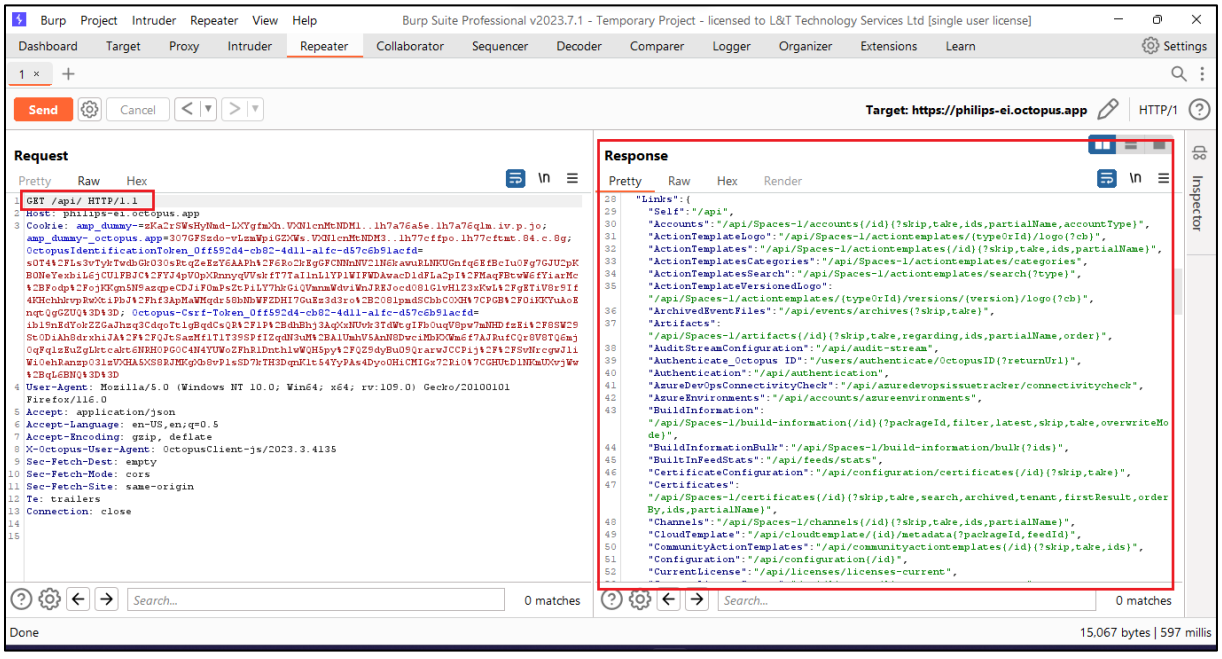
Steps to reproduce:

- Login to the application and launch BurpSuite.
- Capture the request and add "/" after the path and sent the request.
- Here you can observe path are disclosed in the response.

Supportive Evidence:



Exposed sensitive API endpoints.



Exposed sensitive API endpoints.

8.3 WebServices: Broken Authentication via exposed internal API key

Vulnerability Title	Broken Authentication via exposed internal API key
Vulnerability Category	A5- Security Misconfiguration
Severity	**High**
CVSS V3 Calculation	CVSS Base Score: 7.6 CVSS:3.1/AV:N/AC:L/PR:L/UI:N/S:U/C:H/I:L/A:L
Description	**Vulnerability Description:** During the security assessment of the Web application, it is found that the application allows to view the API keys of other users by replacing the user id of victim. **Test as of on 26Jul2023:** Exposed API Keys. **Exploitability Rational:** If an attacker accesses the application, he can be able to view the API keys of victims. **Impact Rational:** Attacker can able get unauthroized access using the victims API keys.
Affected Systems/IP Address/URL	https://Philips-ei.octopus.app GET /api/users/Users-435/apikeys
Recommendation	Prevent authentication information being disclosed such as API key. Revoke access to the exposed API secret keys.
Status	**Open**

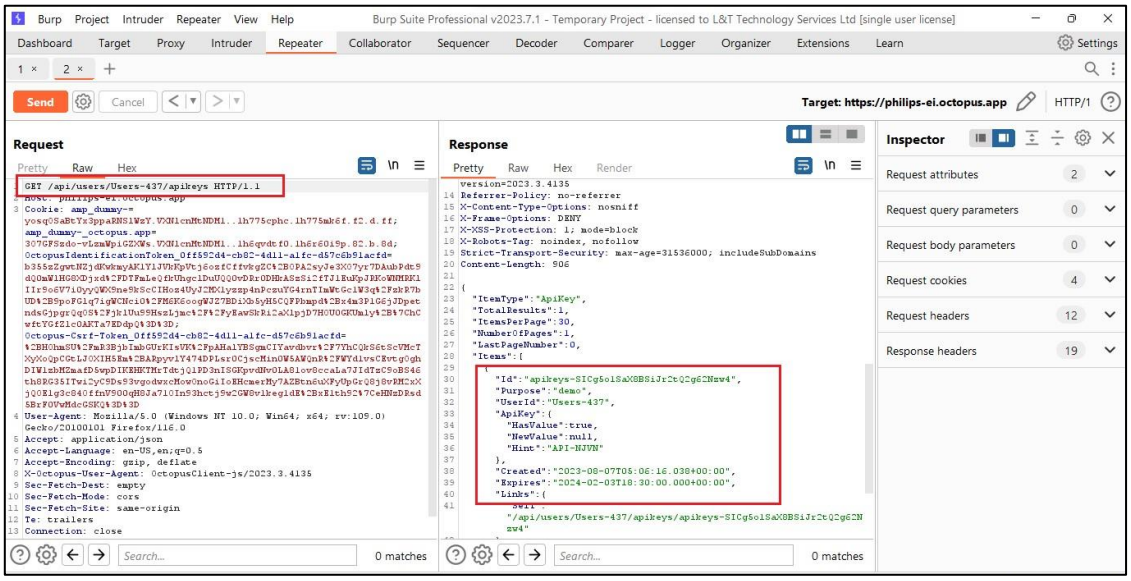
Test as of on 26Jul2023:

Steps to reproduce:

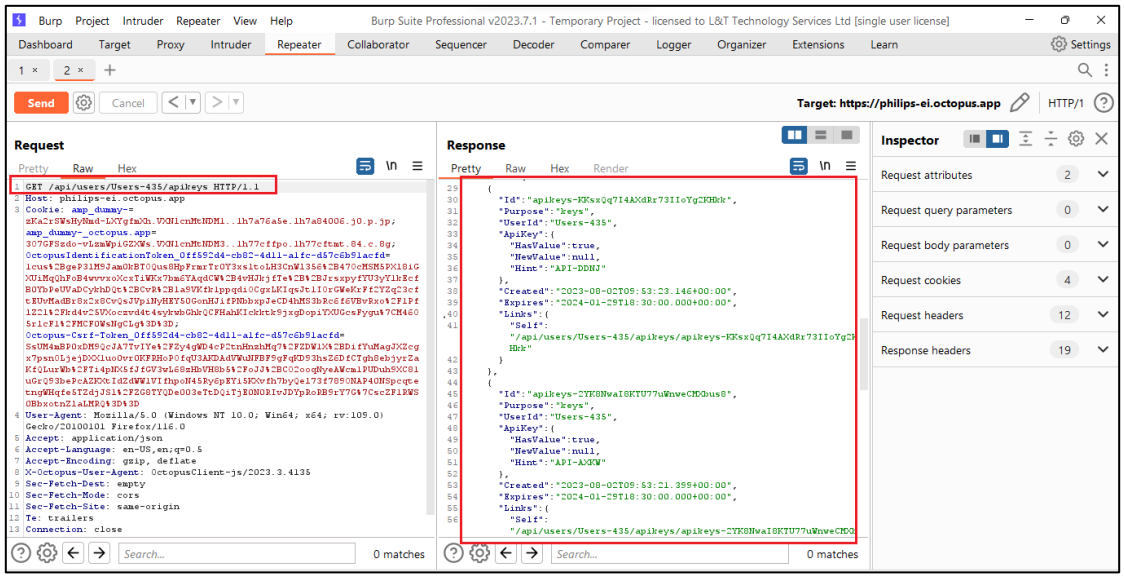
- Login to the application and navigate to profile and click on API keys.
- Capture the request on BurpSuite.
- Here enter the victim user id and trigger the request.

- Now we can able to view the API of victim.

Supportive evidence:



Exposed API Keys for user-437



Exposed API Keys for user-435

8.4 WebServices: API Supports Basic Authentication

Vulnerability Title	API Supports Basic Authentication
Vulnerability Category	API2:2023:Broken Authentication
Severity	**Medium**
CVSS V3 Calculation	CVSS Base Score: 5.3 CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:U/C:N/I:N/A:L
Description	**Vulnerability Description:** It was observed that the web service supports the use of Basic Authentication scheme which is affected by many inherent security flaws. **Test as of on 15Jun2023:** Payload used and it is base64 encoded chars. Authentication mechanism can be bypassed. **Exploitability rational** Basic authentication is vulnerable to replay attacks. Because basic authentication does not encrypt user credentials. It can be easily detected by any sniffing tool. **Impact Rational:** An attacker could intercept traffic between the client and server to review the Basic Auth Header parameter. This header contains the user credentials encoded in Base64 and can be easily decoded to gain application credentials.
Affected Systems/IP Address/URL	https://dev.apps.api.it.philips.com/test/solutionsit/memo-event-mgmt/event https://philipsinformatics-alertsink-dev.eu1.phsdp.com/api/CRM/Case https://philipsinformatics-alertsink-test.eu1.phsdp.com/api/StickyNote
Recommendation	Disable Basic Authentication and use more secure method such as Form Based Authentication.

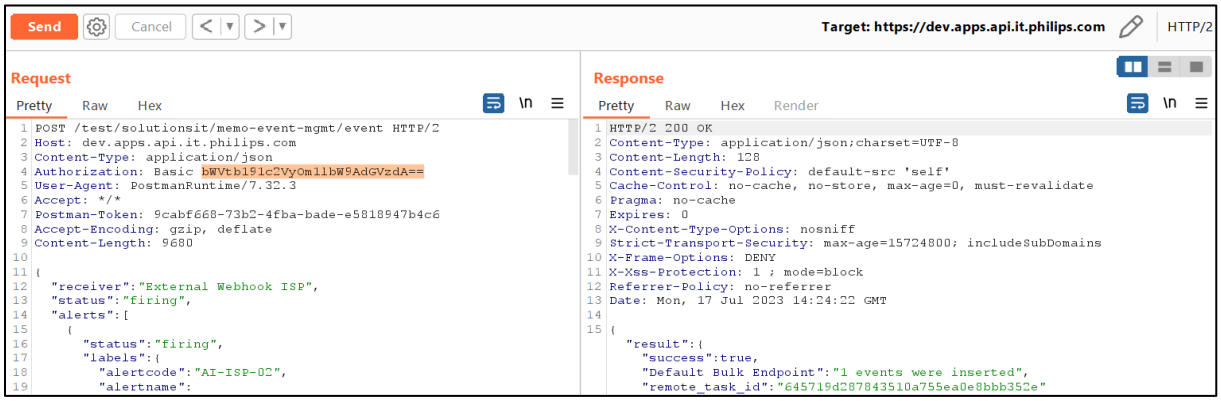
Status	Open

Steps to Reproduce:

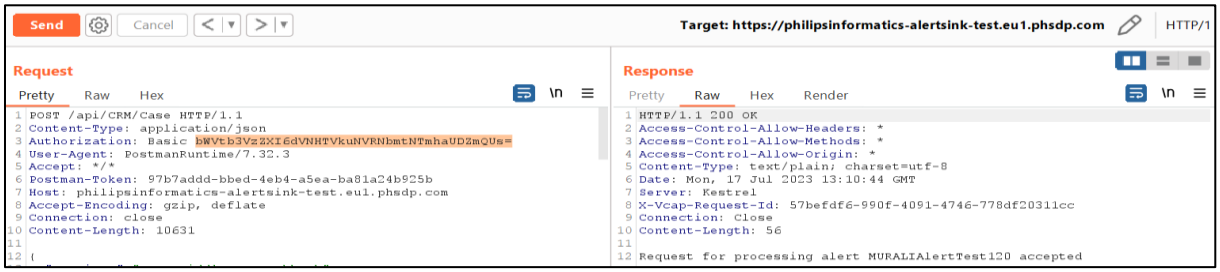
- Load the APIs in the postman & Intercept with burp
- Copy the Authorization Basic value to Burp Decoder -> decode to Base64
- You can able to see the actual value of username/password.

Supportive Evidence:

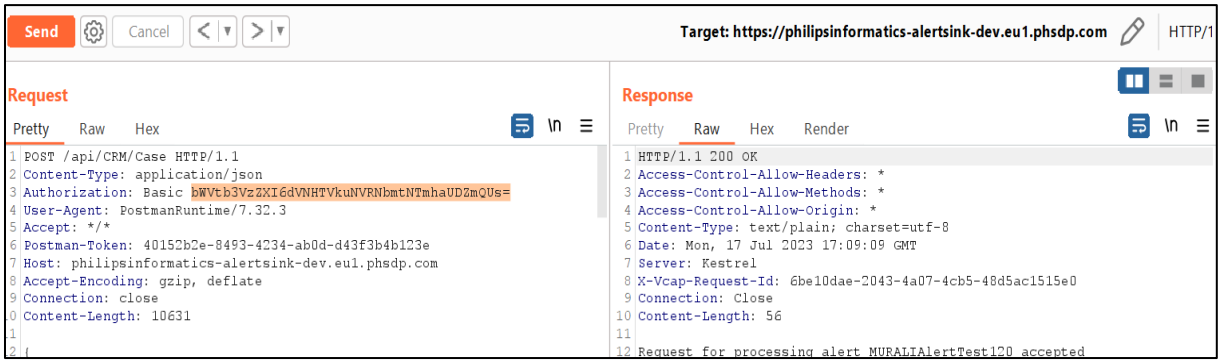
a. Screenshot shows that the Authorization Basic value has been decoded to base64.



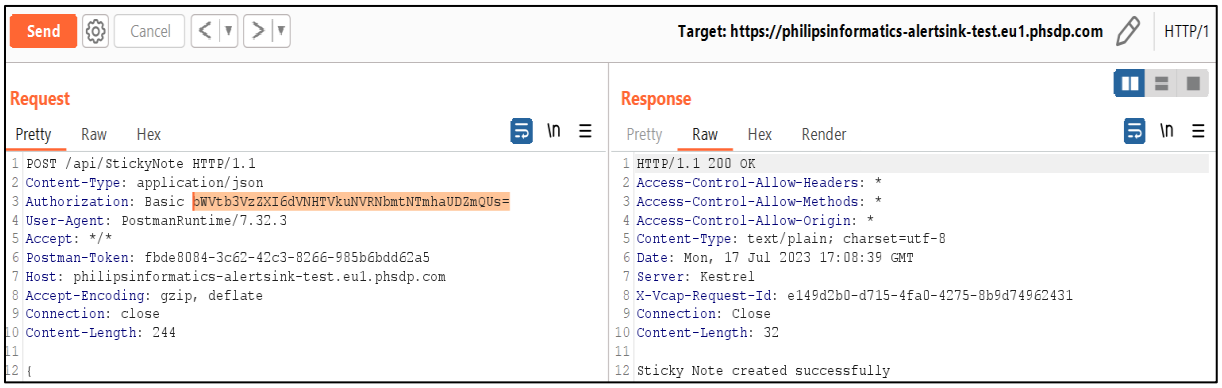
b. Screenshot shows that the Basic authentication used in Memo event mgmt. API.



c. Screenshot shows that the Basic authentication used in CRM Case API



d. Screenshot shows that the Basic authentication used in CRM Case Dev API



e. Screenshot shows that the Basic authentication used in Sticky Note API

8.5 WebServices: CORS Misconfiguration

Vulnerability Title	CORS (Cross Origin Resource Sharing) Misconfiguration
Vulnerability Category	API8:2023- Security Misconfiguration
Severity	**Medium**
CVSS V3 Calculation	CVSS Base Score: 4.7 CVSS:3.0/AV:N/AC:L/PR:N/UI:R/S:C/C:N/I:L/A:N
Description	**Vulnerability Description:** Access-allow control attribute is incorrectly set using wildcards such as (*) under which domains can request resources. It enables controlled access to resources which are outside of the given domain. It adds flexibility to Same Origin Policy (SOP). **Test as of on 15June2023:** We have edited the header and added new header name called Origin and sent a request. We observed that we got 200 response. Origin apart from application should not be allowed. **Exploitability rational:** Malicious websites are able to access and take advantage of the web server's API endpoints due to poor CORS header setting. **Impact Rational:** • This can lead to security misconfiguration. • This is usually set as default, which means any domain can access resources on the site. • Able to steal confidential and sensitive information.
Affected Systems/IP Address/URL	https://philipsinformatics-alertsink-test.eu1.phsdp.com/api/CRM/Case https://philipsinformatics-alertsink-dev.eu1.phsdp.com/api/StickyNote

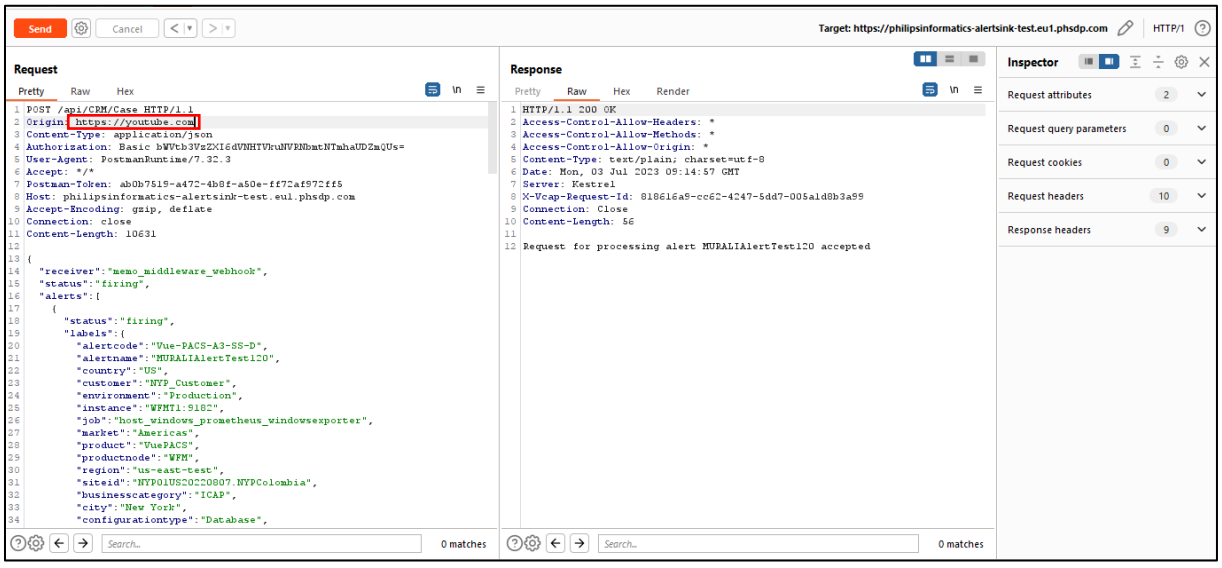
Recommendation	Set the Access-allow control header to validated and whitelisted websites. To implement CORS securely, you need to associate a validation list with Access-Control-Allow-Origin that identifies which specific domains can access resources. Then your application can validate against this list when a domain requests access.
Status	**Open**

Test results as of on 15June2023:

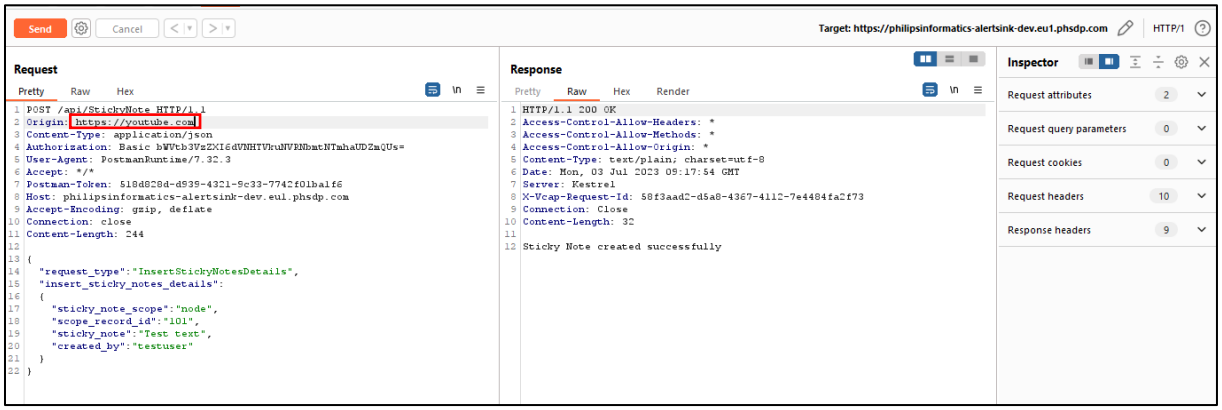
Steps to Reproduce:

- Load the APIs in the postman.
- Edit the headers and add new header name called Origin and send request.
- You can observe that we got 200 response.

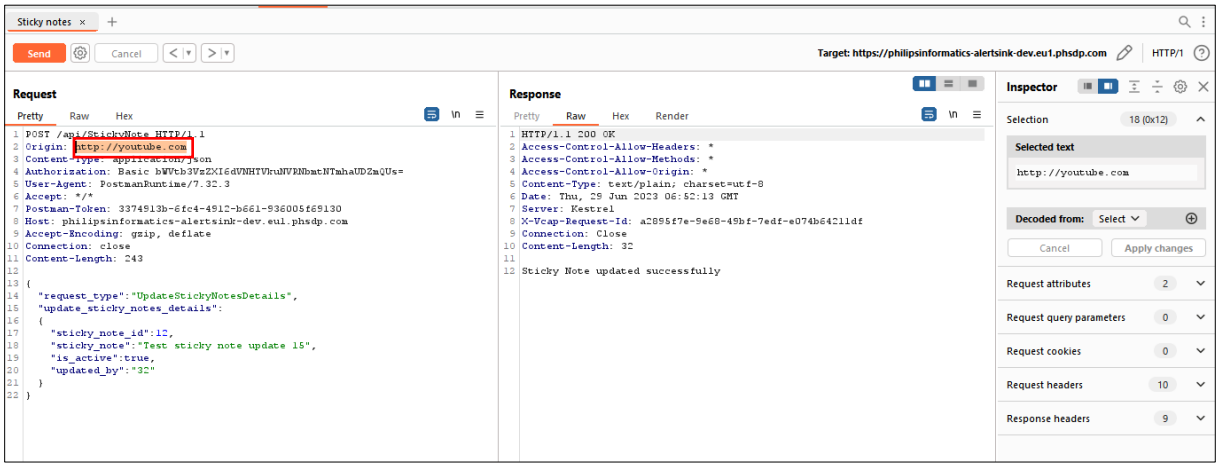
Supportive Evidence:



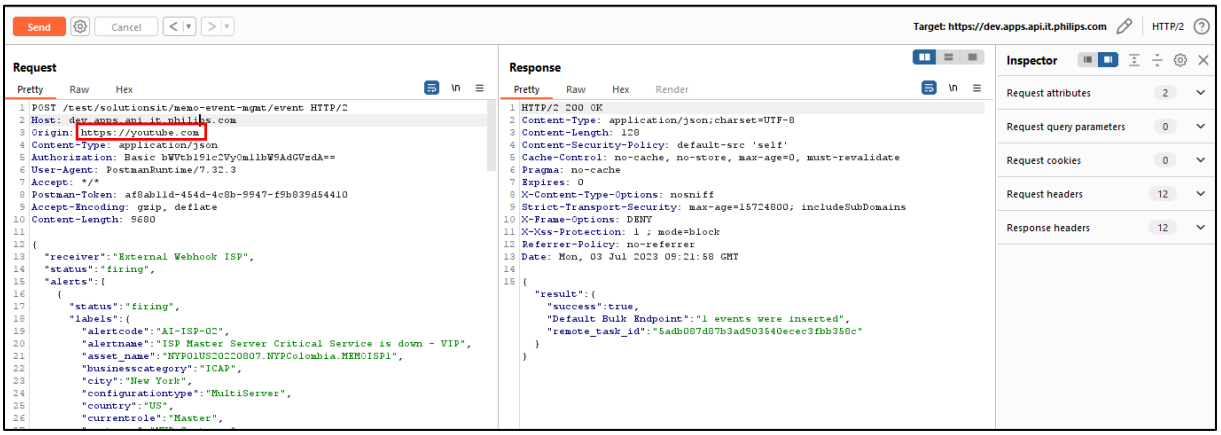
a. **Create Case**



b. **Insert**



c. **Update**



d. **Memo Event**

8.6 DesktopApp: Lack of Binary Protections & Code Signing

Vulnerability Title	Lack of Binary Protection
Vulnerability Category	A5- Security Misconfiguration
Severity	**Medium**
CVSS V3 Calculation	CVSS Base Score: 4.0 CVSS:3.0/AV:L/AC:H/PR:N/UI:N/S:U/C:N/I:L/A:L
Description	**Vulnerability Description:** During the security assessment, it was observed that the Desktop Application binaries like (Memo.ProductExporter, Memo.Promethus, Memo.snmp, Memo.WindowsExporter) and library files are compiled without binary protection flags and lack integrity protection through code signing. - SafeSEH is a security mechanism that helps to block the abuse of SEH based exploitation at runtime. - ControlFlowGuard is a memory protection technique that makes memory corruption and buffer overflow attacks harder to exploit by checking if the jump address is legal or not. - ASLR - randomizes memory addresses at load time, preventing attacks such as return-to-libc that lead to code execution by overwriting specific addresses - DEP - Areas of memory can be marked as non-executive with DEP, preventing an attacker from storing code from buffer overflow attack. - Authenticode or Code Signing – Assemblies can be protected by signing. If left unsigned, an attacker could modify and replace the assembly code with malicious content. **Test as of on 15Jun2023:** All Binaries are unsigned. **Retest as of on 07Aug2023:** Binaries are signed except for product exporter. Hence the issue has been fixed for all the binaries.

	Retest as of on 09Aug2023: Binaries are signed except for product exporter. Hence the issue has been fixed for all the binaries. **Exploitability rational:** Binary exploitation requires advanced skills, and the attacker should have access to machines where the application is deployed. **Impact Rational:** Lack of Binary protection flags makes exploitation of buffer overflow and memory corruption attacks easier on the desktop applications. Attacker can tamper the application binary to include malicious code as the binary is not digitally signed.
Affected Binaries	Philips.MEMO.ProductExporter.exe Philips.MEMO.Prometheus.exe Philips.MEMO.snmp.exe Philips.MEMO.WindowsExporter.exe OctopusTentacle.latest-x64.msi
Recommendation	It is recommended to compile the application with SafeSEH, DEP, ASLR and Control Flow Guard compile time flags enabled and perform code signing on the application binary.
Status	**Open**

Test as of on 15Jun2023:

Steps to Reproduce:

For Binary Protections:

- Download "Get-PESecurity.psm1" tool and run the following command.
- Get-PESecurity -file <path of exe >
- Observe the application binary protection flags are missing.

For Code Signing:

- Download “SysinternalsSuite” tools run the following command.
- sigcheck.exe <path of exe >
- Observe that application is not signed.

Test evidence as of on 15Jun2023:

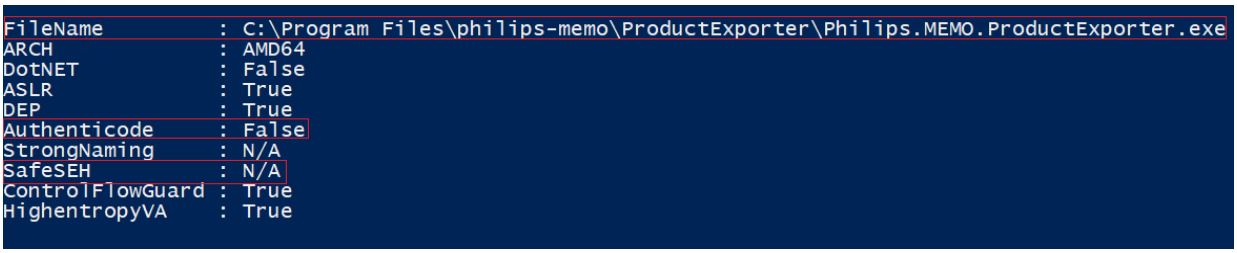
Supportive Evidence:

```
Sigcheck v2.90 - File version and signature viewer
Copyright (C) 2004-2022 Mark Russinovich
Sysinternals - www.sysinternals.com

c:\program files\philips-memo\productexporter\Philips.MEMO.ProductExporter.exe:
        Verified:       Unsigned
        Link date:      11:40 PM 4/13/2022
        Publisher:      n/a
        Company:        Philips
        Description:    Philips.MEMO.ProductExporter
        Product:        Philips.MEMO.ProductExporter
        Prod version:   1.3.0.66
        File version:   1.3.0.66
        MachineType:    64-bit
```

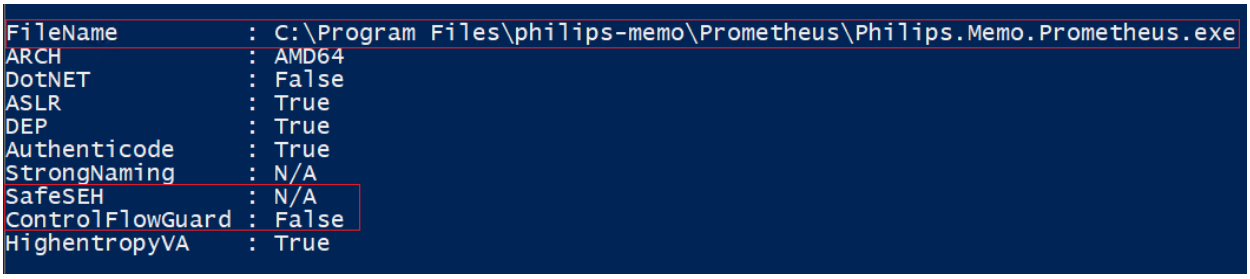
a. Screenshot shows that ProductExporter binary is unsigned

```
Sigcheck v2.90 - File version and signature viewer
Copyright (C) 2004-2022 Mark Russinovich
Sysinternals - www.sysinternals.com

c:\program files\philips snmp\snmp_exporter.exe:
        Verified:       Unsigned
        Link date:      5:30 AM 1/1/1970
        Publisher:      n/a
        Company:        n/a
        Description:    n/a
        Product:        n/a
        Prod version:   n/a
        File version:   n/a
        MachineType:    64-bit
```

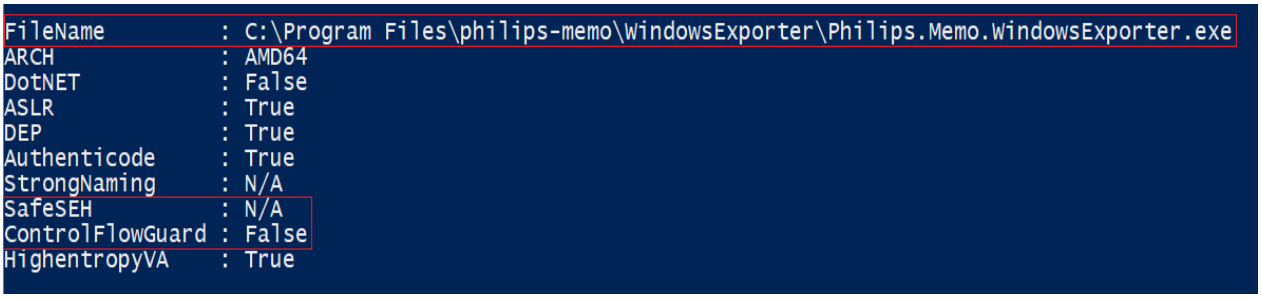
b. Screenshot shows that snmp.exporter binary is unsigned



c. Screenshot shows that ProductExporter binary does not implement Authenticode & SafeSEH



d. Screenshot shows that Prometheus binary does not implement SafeSEH & ControlFlowGuard

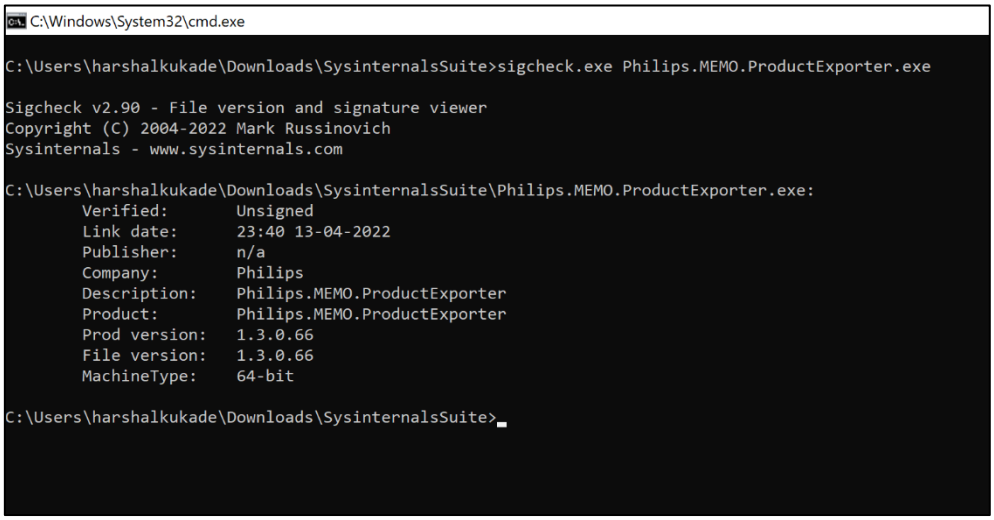


e. Screenshot shows that Windows Exporter binary does not implement SafeSEH & ControlFlowGuard

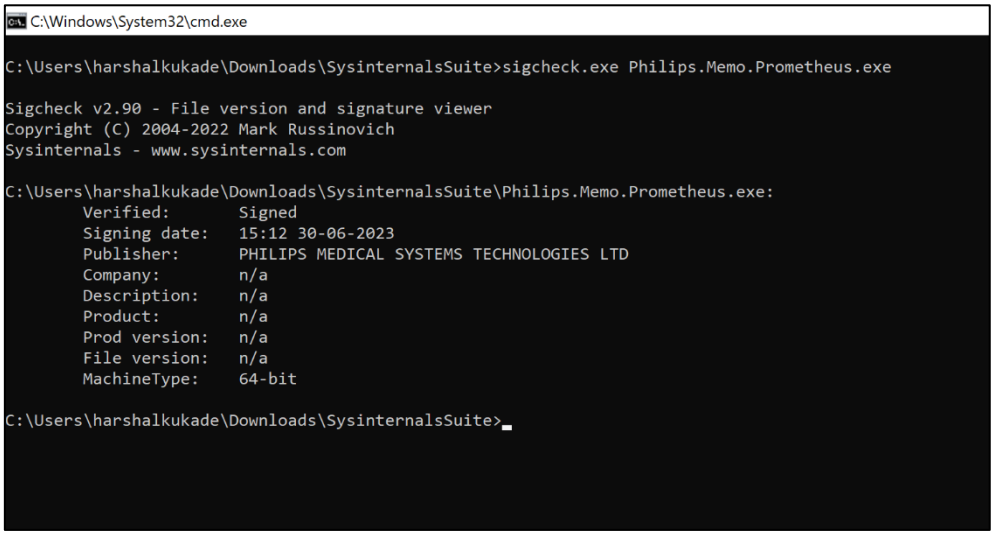
```
FileName          : C:\Program Files\Philips SNMP\snmp_exporter.exe
ARCH              : AMD64
DotNET            : False
ASLR              : True
DEP               : True
Authenticode      : False
StrongNaming      : N/A
SafeSEH           : N/A
ControlFlowGuard  : False
HighentropyVA     : True
```

f. Screenshot shows that snmp exporter binary does not implement Authenticode SafeSEH & ControlFlowGuard

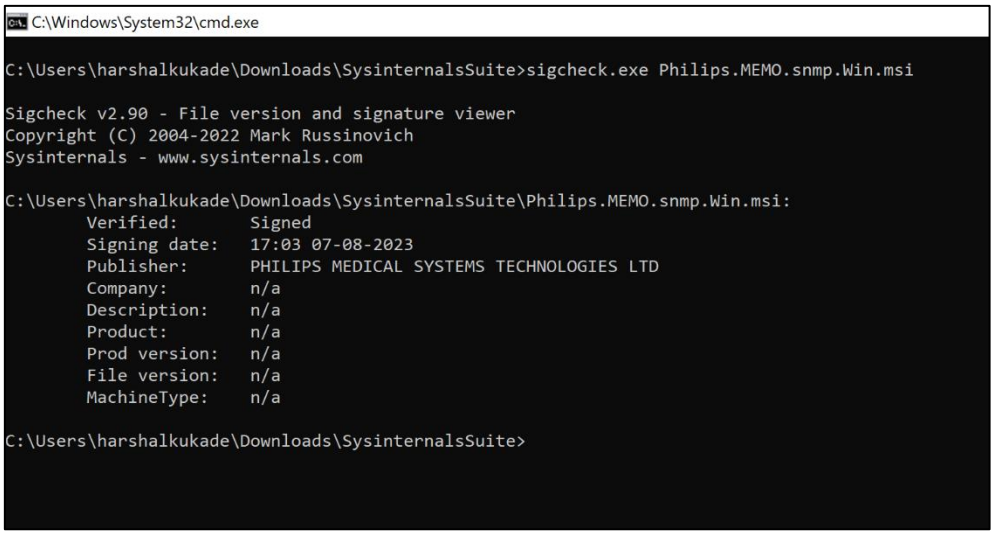
Test evidence as of on 07Aug2023:



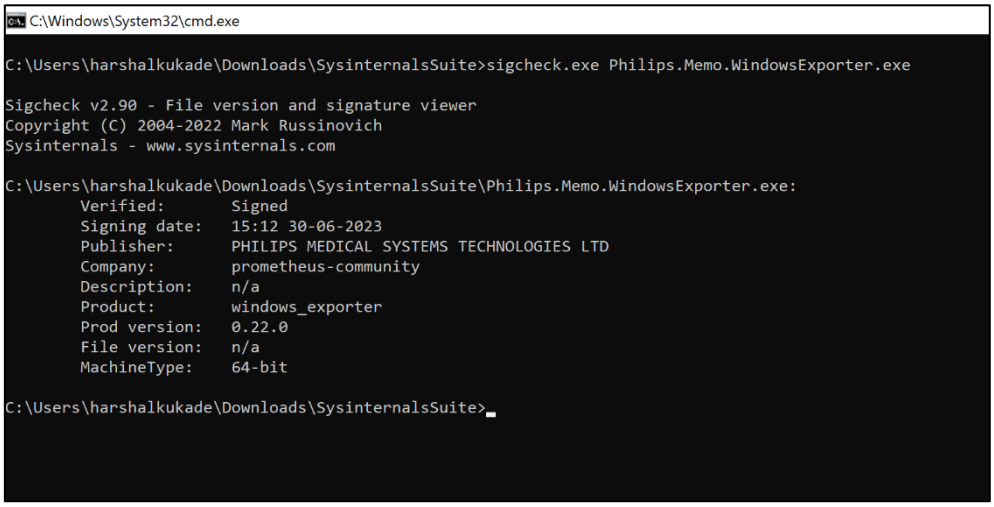
ProductExporter binary verified as unsigned.



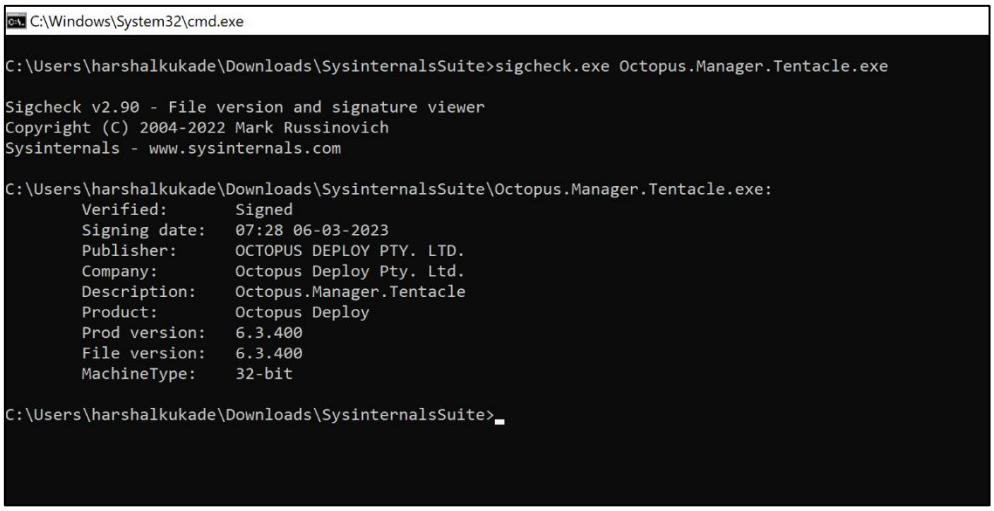
Prometheus binary verified as signed.



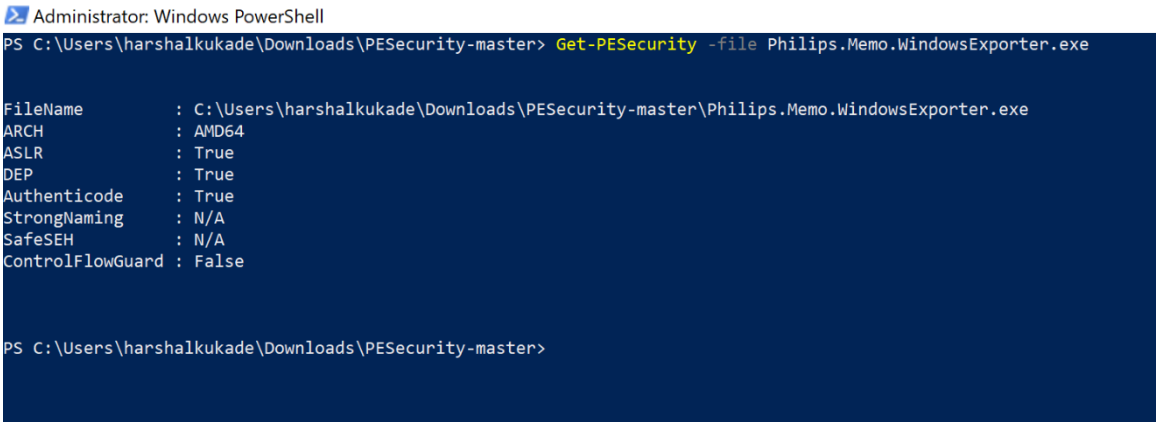
Snmp binary verified as signed.



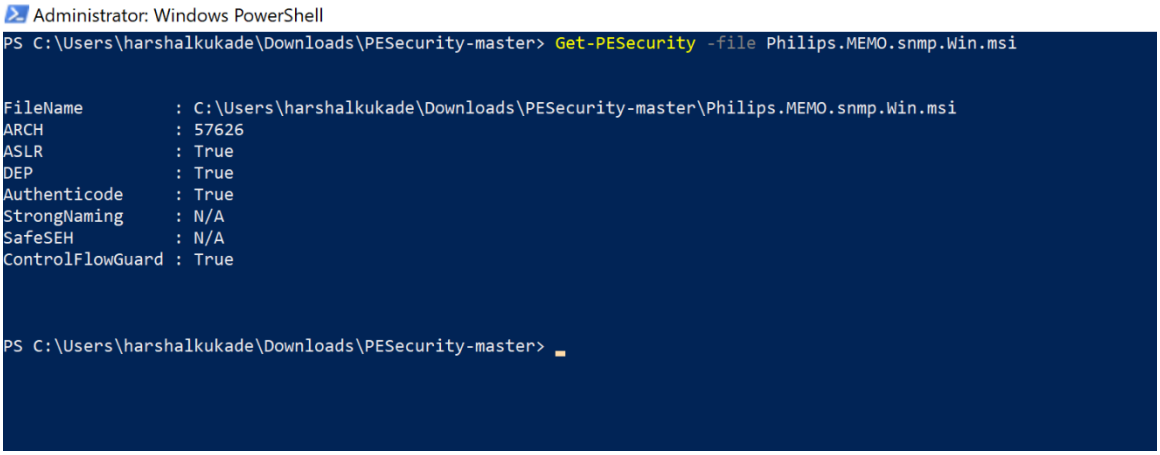
WindowsExporter binary verified as signed.

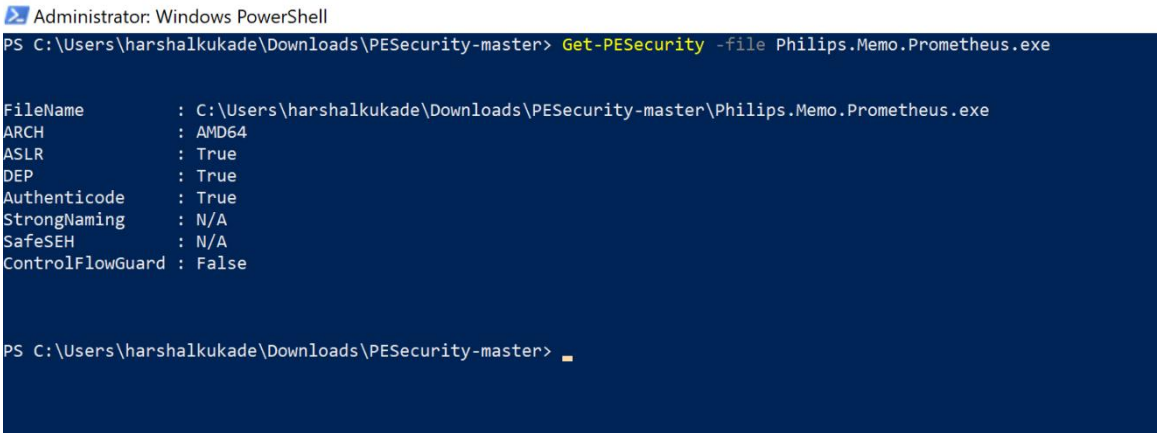


Octopus binary verified as signed.

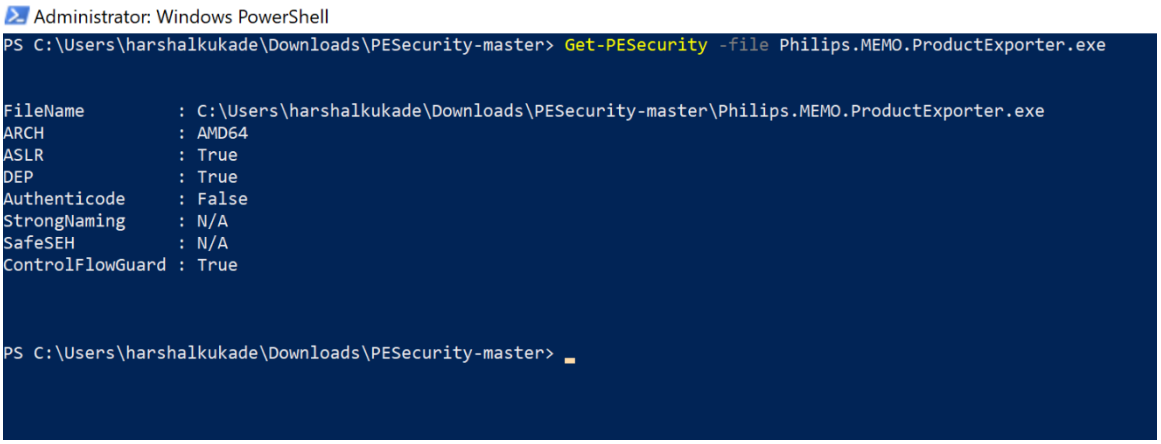


SafeSEH is NA and CFG is False

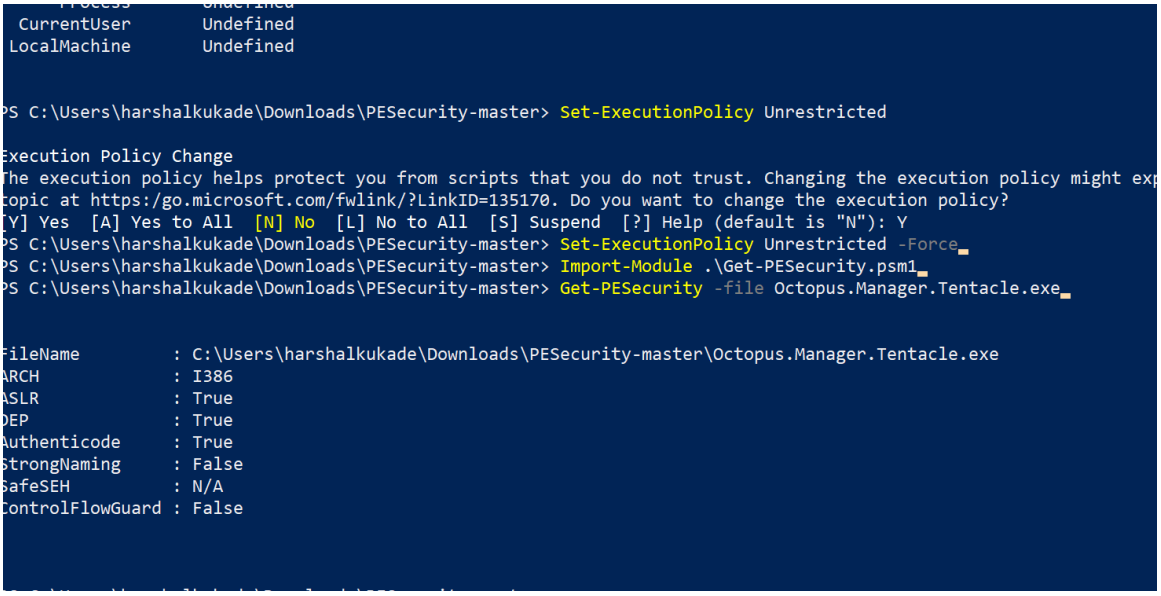




CFG is False

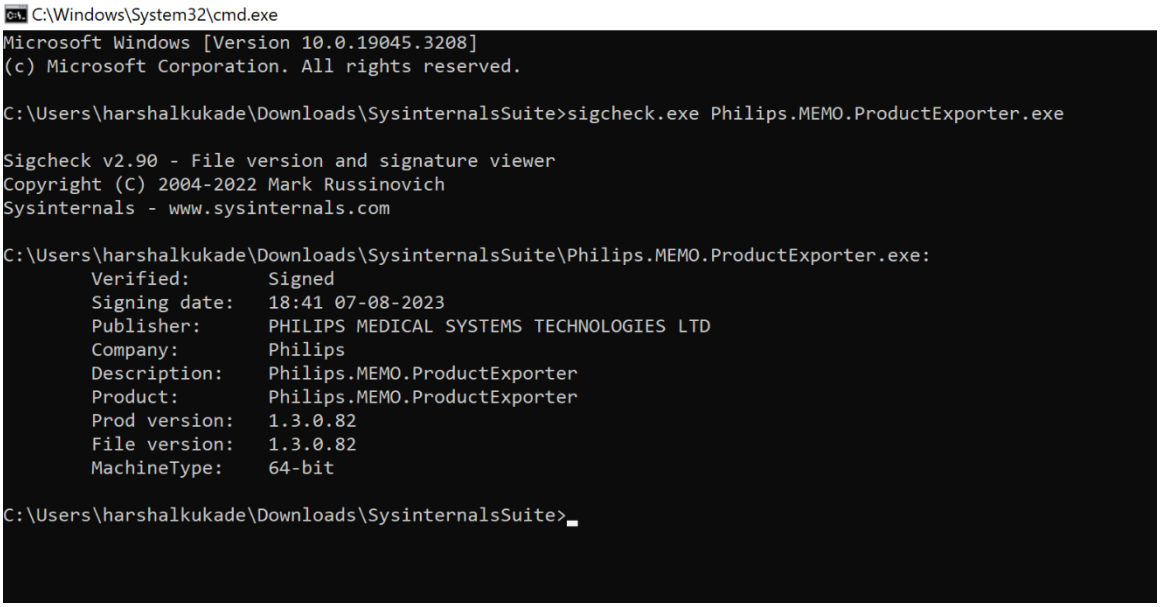


Authenticode is False

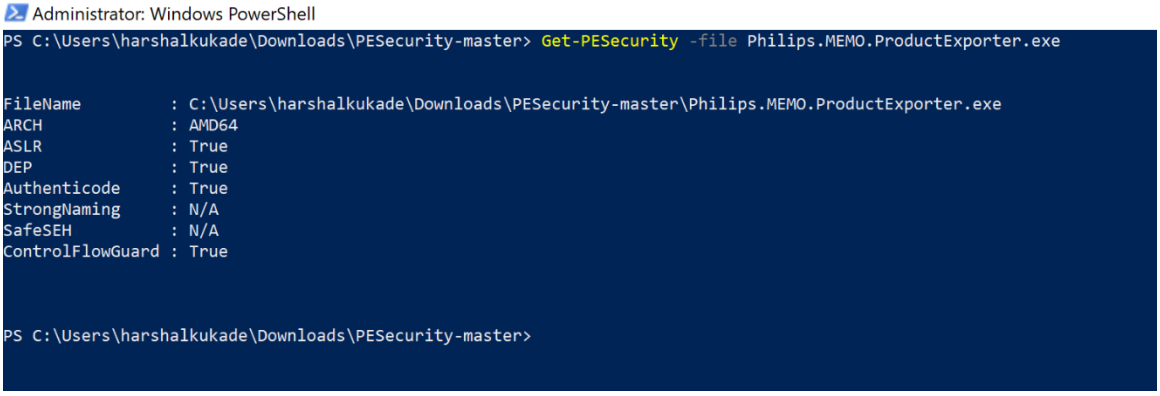


CFG is False

Retest as of on 09Aug2023



ProductExporter exe is signed. Issue fixed.



Product Exporter exe Authenticode is True.

Binary/DesktopApp	Signature Verification	Authenticode	SafeSEH	CFG
Philips.MEMO.ProductExporter.exe	signed	FALSE	NA	TRUE
Philips.MEMO.Prometheus.exe	signed	TRUE	NA	FALSE
Philips.MEMO.snmp.exe	signed	FALSE	NA	FALSE
Philips.MEMO.WindowsExporter.exe	signed	TRUE	NA	FALSE
OctopusTentacle.latest-x64.msi	signed	TRUE	NA	FALSE

8.7 DesktopApp: Information Disclosure

Vulnerability Title	Information Disclosure
Vulnerability Category	A5- Security Misconfiguration
Severity	**Medium**
CVSS V3 Calculation	CVSS Base Score: 5.3 CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:U/C:L/I:N/A:N
Description	**Vulnerability Description:** It was observed that the application discloses software and technical details through HTTP response. Following information were disclosed: • Microsoft-HTTPAPI/2.0 • Build User Name - runneradmin & root • go1.20.2 • go1.20.4 **Test as of on 15Jun2023:** Application is disclosing version details and hence we kept the issue as open. **Exploitability rational:** An attacker can use this information to enhance their understanding on the application attack surface and research attack vectors. **Impact Rational:** This information available could be used to attempt more sophisticated attacks against the application, by understanding backend frameworks and internal details of the application.
Affected Hosts	http://memoconc:9925/metrics1s

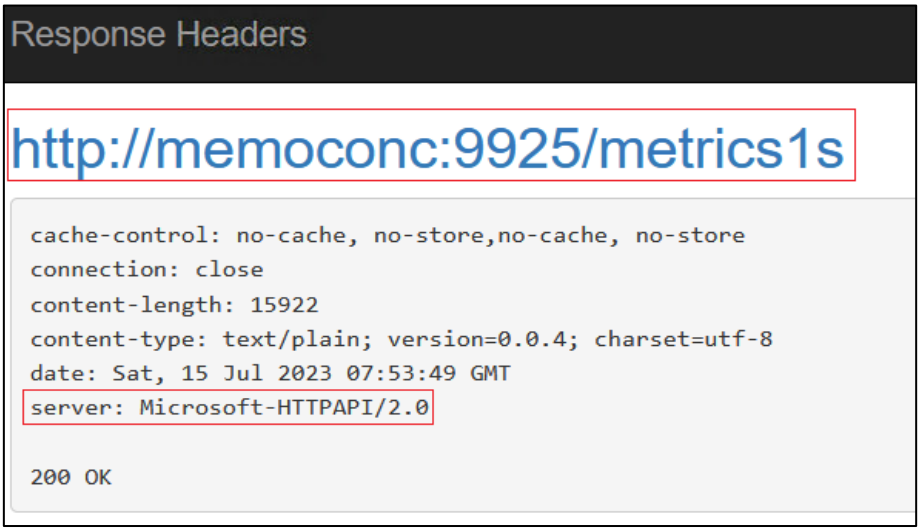
	http://memoconc:9182/version http://memoconc:9090/api/v1/status/buildinfo
Recommendation	It is recommended to remove all information like server/framework type and versions from all response headers and service banners.
Status	**Open**

Test as of on 15Jun2023:

Steps to Reproduce:

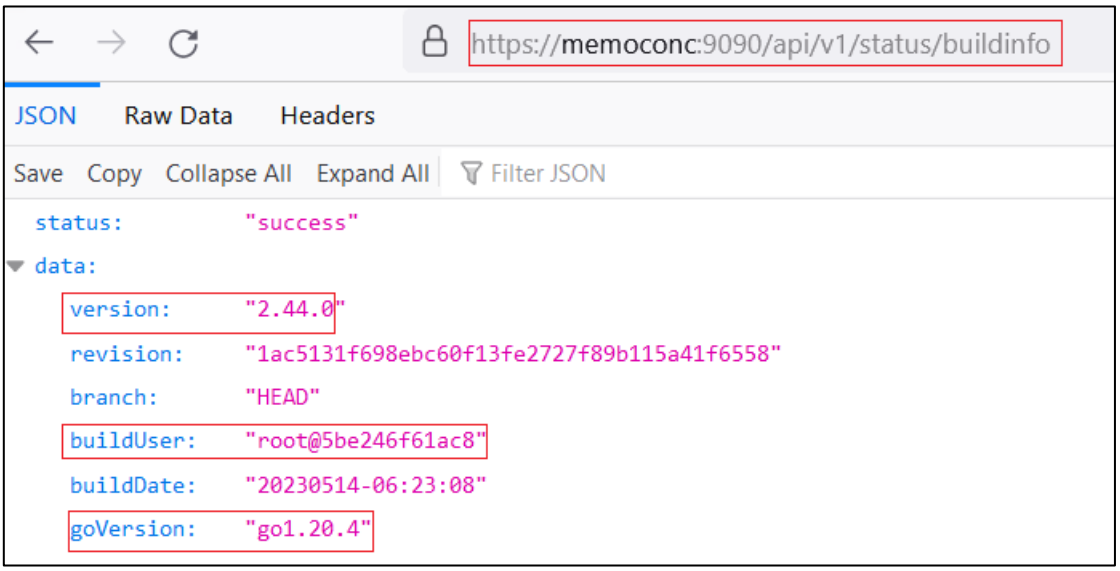
- Visit the affected endpoints and view the response header
- Observe that the application discloses version details.

Supportive Evidence:



a. Screenshot shows that the application discloses Microsoft-HTTPAPI/2.0 version

b. Screenshot shows that the application discloses build User name & go Version



c. Screenshot shows that the application discloses build User name & go Version

8.8 DesktopApp: Insecure HTTP Methods Allowed

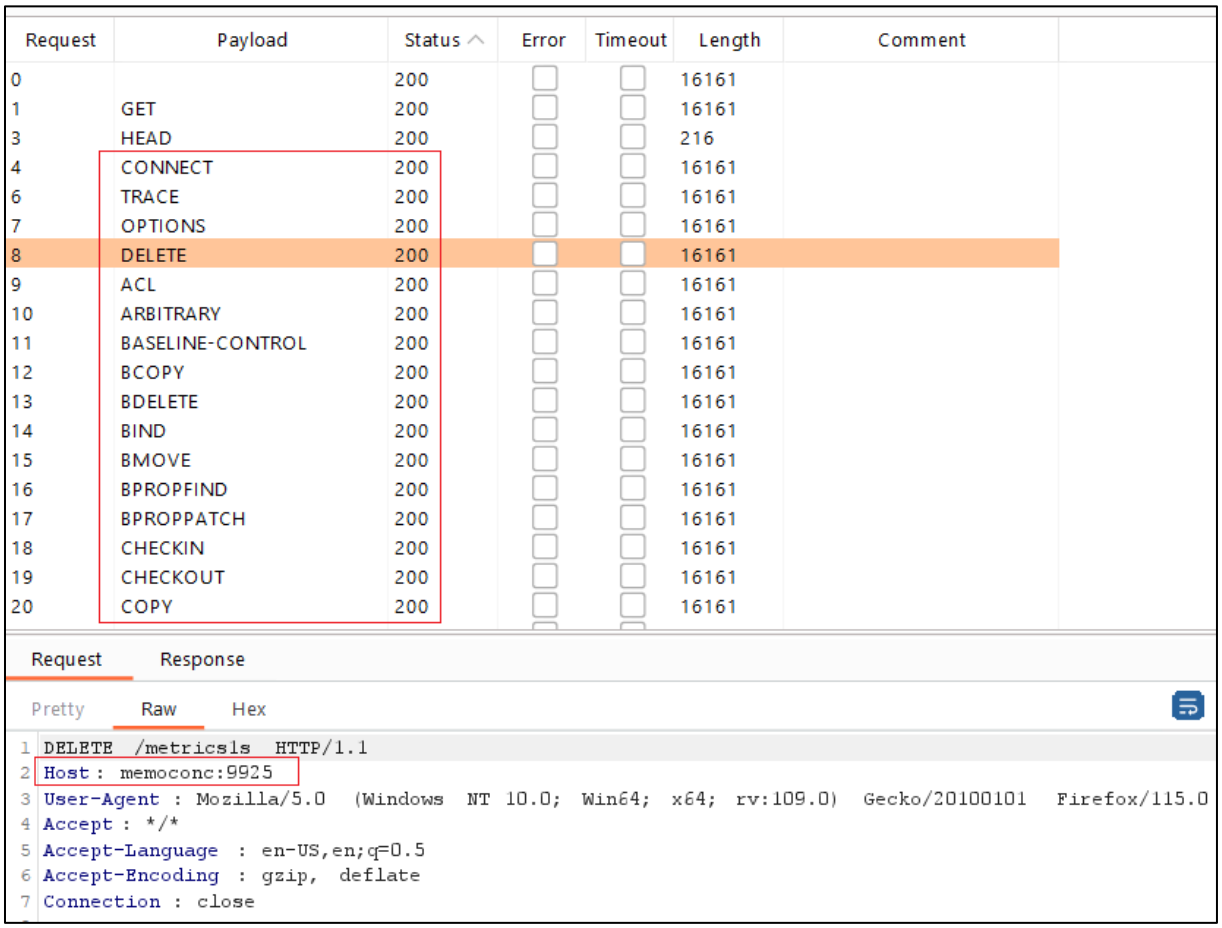
Vulnerability Title	Insecure HTTP Methods Allowed
Vulnerability Category	A5- Security Misconfiguration
Severity	**Medium**
CVSS V3 Calculation	CVSS Base Score: 5.3 CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:U/C:L/I:N/A:N
Description	**Vulnerability Description:** It was observed that the application server supports most of the Insecure HTTP methods allowed like PUT, DELETE, TRACE, CONNECT, DEBUG, PATCH etc. **Test as of on 15Jun2023:** Attacker is able to request with Insecure HTTP methods in the application with 200 OK response. **Exploitability rational:** It is relatively difficult to exploit the insecure http methods, since it requires platform level authentication to start a remote session and launch attacks against the system. **Impact Rational:** If an attacker can successfully start a remote debugging session, this is likely to disclose sensitive information about the application and supporting infrastructure that may be valuable in formulating targeted attacks against the system.
Affected Hosts	http://memoconc:9925/metrics1s
Recommendation	It is recommended to disable all unnecesary HTTP Methods on the server.
Status	**Open**

Test as of on 15Jun2023:

Steps to Reproduce:

- Visit the affected endpoint in the browser
- Intercept with burp suite & send the request to Intruder
- Choose HTTP Method as payload position, choose sniper attack, in payload list Use HTTP Verbs and start the Intruder attack
- Observe that application responded with 200ok for all the Insecure HTTP Methods.

Supportive Evidence:



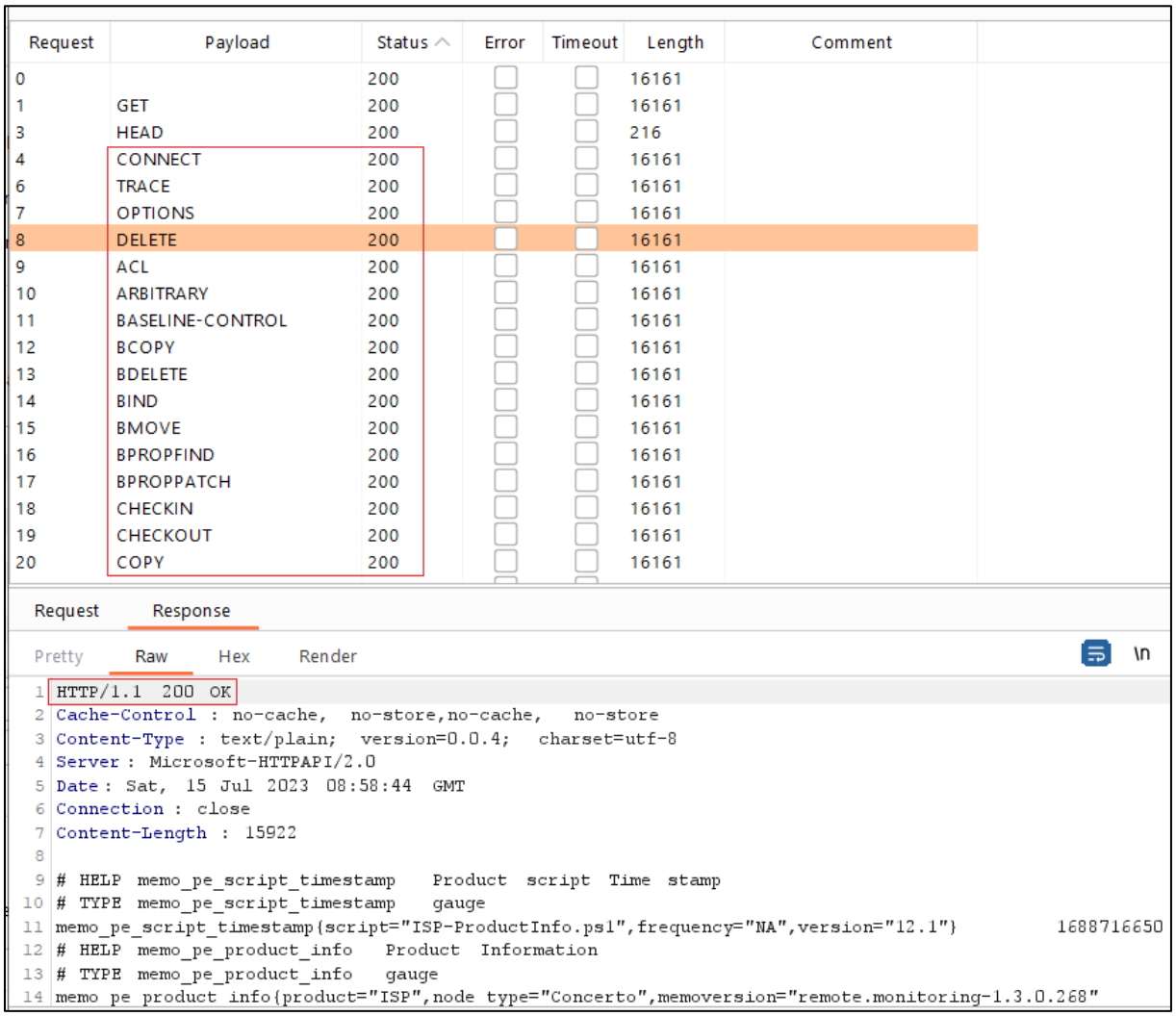
Request	Payload	Status	Error	Timeout	Length	Comment
0		200			16161	
1	GET	200			16161	
3	HEAD	200			216	
4	CONNECT	200			16161	
6	TRACE	200			16161	
7	OPTIONS	200			16161	
8	DELETE	200			16161	
9	ACL	200			16161	
10	ARBITRARY	200			16161	
11	BASELINE-CONTROL	200			16161	
12	BCOPY	200			16161	
13	BDELETE	200			16161	
14	BIND	200			16161	
15	BMOVE	200			16161	
16	BPROPFIND	200			16161	
17	BPROPPATCH	200			16161	
18	CHECKIN	200			16161	
19	CHECKOUT	200			16161	
20	COPY	200			16161	

Request Response

Pretty Raw Hex

```
1 DELETE /metricsls HTTP/1.1
2 Host : memoconc:9925
3 User-Agent : Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:109.0) Gecko/20100101 Firefox/115.0
4 Accept : */*
5 Accept-Language : en-US,en;q=0.5
6 Accept-Encoding : gzip, deflate
7 Connection : close
```

a. Screenshot shows that the attacker has requested with Insecure HTTP methods in the application



Request	Payload	Status	Error	Timeout	Length	Comment
0		200			16161	
1	GET	200			16161	
3	HEAD	200			216	
4	CONNECT	200			16161	
6	TRACE	200			16161	
7	OPTIONS	200			16161	
8	DELETE	200			16161	
9	ACL	200			16161	
10	ARBITRARY	200			16161	
11	BASELINE-CONTROL	200			16161	
12	BCOPY	200			16161	
13	BDELETE	200			16161	
14	BIND	200			16161	
15	BMOVE	200			16161	
16	BPROPFIND	200			16161	
17	BPROPPATCH	200			16161	
18	CHECKIN	200			16161	
19	CHECKOUT	200			16161	
20	COPY	200			16161	

Request Response

Pretty Raw Hex Render \n

```
1 HTTP/1.1 200 OK
2 Cache-Control : no-cache, no-store,no-cache, no-store
3 Content-Type : text/plain; version=0.0.4; charset=utf-8
4 Server : Microsoft-HTTPAPI/2.0
5 Date : Sat, 15 Jul 2023 08:58:44 GMT
6 Connection : close
7 Content-Length : 15922
8
9 # HELP memo_pe_script_timestamp Product script Time stamp
10 # TYPE memo_pe_script_timestamp gauge
11 memo_pe_script_timestamp{script="ISP-ProductInfo.ps1",frequency="NA",version="12.1"} 1688716650
12 # HELP memo_pe_product_info Product Information
13 # TYPE memo_pe_product_info gauge
14 memo_pe_product_info{product="ISP",node type="Concerto",memoversion="remote.monitoring-1.3.0.268"
```

b. Screenshot shows that the application responded back with 200 ok

8.9 DesktopApp: TLS Implementation Flaws

Vulnerability Title	TLS Implementation Flaws
Vulnerability Category	A2 – Cryptographic Issues
Severity	**Medium**
CVSS V3 Calculation	CVSS Base Score: 5.3 CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:U/C:L/I:N/A:N
Description	**Vulnerability Description:** It was observed that the remote server allows clients to use TLSv1.2 protocols. The following deficiencies were found in the encrypted communication Configuration: - Sweet32 – It uses Collision or "birthday" attack against 64-bit DES/3DES ciphers in CBC mode to decrypt sensitive information like session cookie, by sending large amount of data over a single SSL/TLS session. - Lucky13 – It was observed that the remote server is vulnerable to LUCKY13 attack. The Lucky Thirteen attack Is a cryptographic timing attack against implementations of the transport layer security (TLS) protocol that use the CBC mode of operation. **Test as of on 15Jun2023:** Application is using CBC Weak cipher under TLSv1.2. **Exploitability rational:** When weak cipher suites such as CBC and 3DES are configured for the webserver, it leaves the application vulnerable to timing and MITM attacks. However, it seems difficult to exploit. **Impact Rational:** - **Sweet32** – An attacker who can send arbitrary HTTP Requests on behalf of the user by controlling the client and can sniff the HTTPS

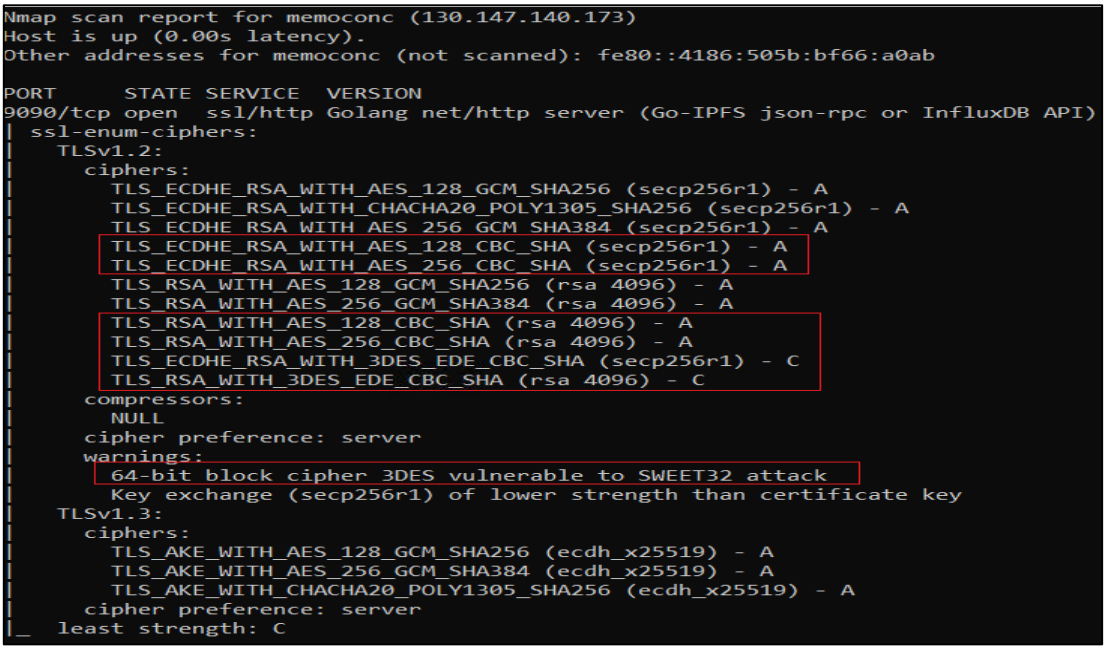
	response, is then able to decrypt one block of the encrypted message within 232 attacker-controlled requests. - **Lucky13 –** An attacker would measure the time it takes to encrypt records when using the standard CBC cipher suites used in TLS Sessions. An attacker would then be able to recover plain text from TLS connections that use CBC mode encryption.
Affected Hosts	http://memoconc:9182/ http://memoconc:9090/
Recommendation	It is recommended to disable support of Cryptographic protocols with known vulnerabilities and the server should enforce the use of latest stable version of TLS. In addition weak or lowgrade CBC ciphers or encryption must be disabled on the application server.
Status	**Open**

Test as of on 15Jun2023:

Steps to Reproduce:

- Run nmap command line: nmap -sV --script ssl-enum-ciphers <host>
- Observe the result that application uses weak ciphers.

Supportive Evidence:



a. Screenshot shows that the application using Weak ciphers.

```
Nmap scan report for memoconc (130.147.140.173)
Host is up (0.0010s latency).
Other addresses for memoconc (not scanned): fe80::4186:505b:bf66:a0ab

PORT     STATE SERVICE
9182/tcp open  unknown
| ssl-enum-ciphers:
|   TLSv1.2:
|     ciphers:
|       TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 (secp256r1) - A
|       TLS_ECDHE_RSA_WITH_CHACHA20_POLY1305_SHA256 (secp256r1) - A
|       TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 (secp256r1) - A
|       TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA (secp256r1) - A
|       TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA (secp256r1) - A
|       TLS_RSA_WITH_AES_128_GCM_SHA256 (rsa 4096) - A
|       TLS_RSA_WITH_AES_256_GCM_SHA384 (rsa 4096) - A
|       TLS_RSA_WITH_AES_128_CBC_SHA (rsa 4096) - A
|       TLS_RSA_WITH_AES_256_CBC_SHA (rsa 4096) - A
|       TLS_ECDHE_RSA_WITH_3DES_EDE_CBC_SHA (secp256r1) - C
|       TLS_RSA_WITH_3DES_EDE_CBC_SHA (rsa 4096) - C
|     compressors:
|       NULL
|     cipher preference: server
|     warnings:
|       64-bit block cipher 3DES vulnerable to SWEET32 attack
|       Key exchange (secp256r1) of lower strength than certificate key
|   TLSv1.3:
|     ciphers:
|       TLS_AKE_WITH_AES_128_GCM_SHA256 (ecdh_x25519) - A
|       TLS_AKE_WITH_AES_256_GCM_SHA384 (ecdh_x25519) - A
|       TLS_AKE_WITH_CHACHA20_POLY1305_SHA256 (ecdh_x25519) - A
|     cipher preference: server
|_  least strength: C
```

b. Screenshot shows that the application using Weak ciphers.

8.10 WebApp: Weak Password Complexity Requirements

Vulnerability Title	Weak Password Policy
Vulnerability Category	A7- Identification and Authentication Failures
Severity	**Medium**
CVSS V3 Calculation	CVSS Base Score: 5.3 CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:N/I:L/A:N
Description	**Vulnerability Description:** During the security assessment, it is observed that the application does not follow password policy. Weak password policy raise the possibility that users may use weak passwords, which makes it simpler for attackers to obtain user passwords using common attack methods (such as brute force attacks, authentication theft). **Test as of on 15June2023:** Application is supporting weak passwords under authentication functionality. **Exploitability rational:** Without password rules, users are likely to use weak passwords, and there may be problems with reliability and authentications. For exploiting this vulnerability, the attacker needs to know the User's old password to brute force via login functionality. **Impact Rational:** • Easily guessable and are easy target for brute force attacks. • If the old password was revealed to someone else (can happen, especially with phishing) then there is a chance to login with the credentials and can gain access to the application. • It is possible someone may gain access to your saved passwords. • It can lead to authentication failure. • Weak password complexity requirements makes the application vulnerable to bruteforce attacks, which may potentially lead to compromise of the user accounts. Attackers that gain access to

	these user accounts may be able to perform fraudulent actions, as well as gaining access to sensitive information.
Affected Systems/IP Address/URL	https://mas-edi-grafana-client-test.eu-west.monitoring.hsdp.io/profile/password https://philips-ei.octopus.app/app#/
Recommendation	Strong password with more than 8 characters (combination of uppercase letters, lowercase letters, numbers and symbols) should be required to provide adeqate security. It is recommended that the application should enforce password complexity requirements that are in accordance with Philips password policy.
Status	**Open**

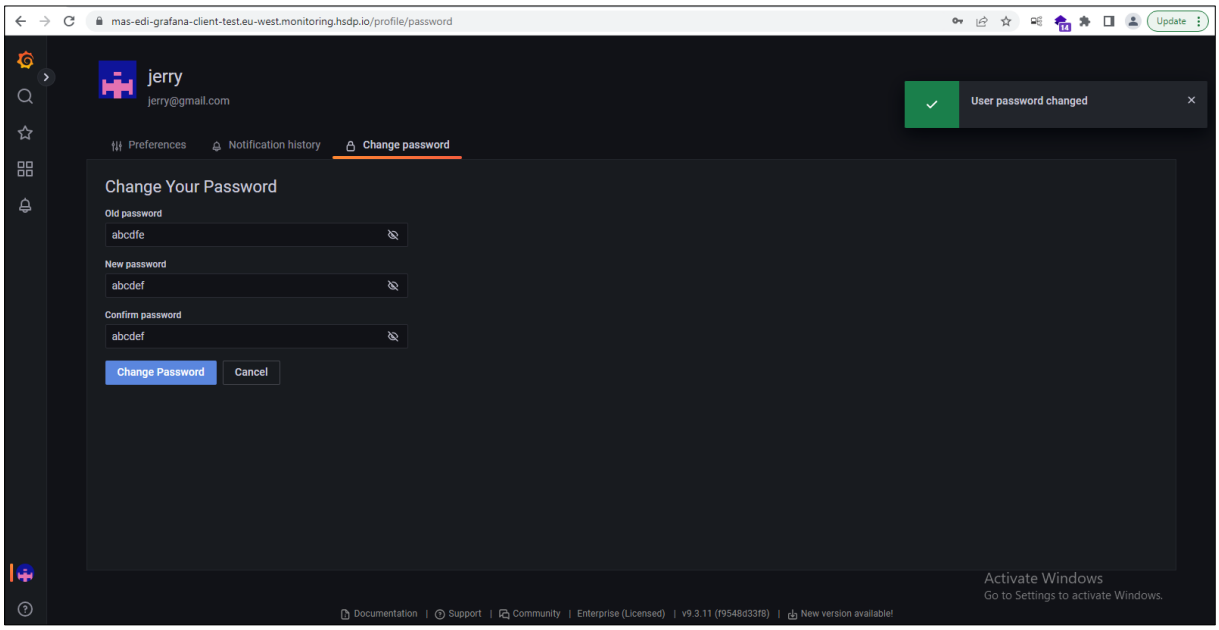
Case 1: Grafana

Test as of on 15June2023:

Steps to Reproduce:

- Open the application in the browser.
- Go to profile and select change password.
- While changing the password we can observe that it's taking weak password (less than 8 characters).

Supportive Evidence:



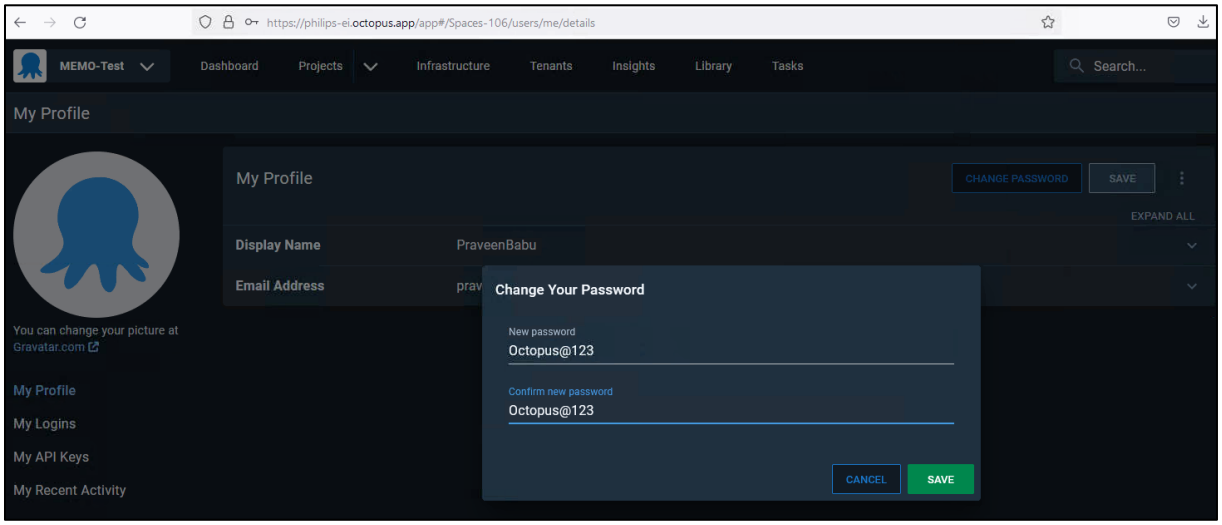
Case 2: Octopus

Test as of on 01Aug2023:

Steps to Reproduce:

1) Login to https://philips-ei.octopus.app/app#/ with valid credentials.
2) Navigation to My profile section.
3) Click on Change your password option.
4) Provide the old as password as new password & save it.
5) Application will accept the same.

Supportive Evidence:



Screenshot shows that the Octopus application accepting the old password as new password.

8.11 Webapp: Improper Session Management

Vulnerability Title	Improper Session Management
Vulnerability Category	A7 Identification and Authentication Failures
Severity	**Medium**
CVSS V3 Calculation	CVSS Base Score: 6.1 CVSS:3.1/AV:A/AC:H/PR:L/UI:R/S:U/C:H/I:H/A:N
Description	**Vulnerability Description:** **Case 1:** It was observed that session id is not invalidated at the server side due to which session id of one user can be used for the other user. **Case 2:** The application allows concurrent sessions: It was observed that upon deletion of user account, if the user is already logged in to the application via different browser, the existing session remain alive without any restrictions. Once the user logs out, then that user needs to sign up again to access the application. But until then that user can access and perform all operations. **Exploitability rational** For the case 1, attacker should have access to the application. For the case 2, user needs to be logged in to the application. For the case 3, user need to be logged in to the application in two or more different tabs of same browser window. For the case 4 and case 5 to be exploited, the attacker needs physical access to the system. **Impact Rational:** Leaving the user's session active after the user initiates logout provides the attacker with a larger window in which to steal a victim's session and impersonate that user in the application. The session token can be obtained through various techniques, such as intercepting network traffic (“Man-in-the-Middle” or “MitM”), Cross-Site Scripting (XSS), Cross-Site Tracing (XST), and in the case of URL-based session token transmission, local inspection of browser history. In addition to prolonging the session identifier's exposure to attack, failing to invalidate the user's session server-side also leaves the user with no way to deny an attacker's access once the victim discovers that their session has been compromised.

Affected Systems/IP Address/URL	https://philips-ei.octopus.app/app#/
Recommendation	The user's HTTP session should be terminated on the server immediately after a logout action is performed. It is important to note that simply deleting the cookie from the browser will not terminate the server session. The session must be invalidated at the server, using the HTTP container's intrinsic session abandonment mechanism. Reference: https://cheatsheetseries.owasp.org/cheatsheets/Session_Management_Cheat_Sheet.html
Status	**Open**

Test as of on 01Aug2023:

Steps to Reproduce:

Case 1:

Step 1: In browser 1, log in to the application as user A.

Step 2: Using Cookie editor, Export the cookies & Copy to clipboard & signout from Browser 1

Step 3: In browser 2, visit the app url, it will open the login page

Step 4: Using Cookie editor, Import the cookies & Save it

Step 5: Now refresh the browser2, application will login without any credentials.

Case 2:

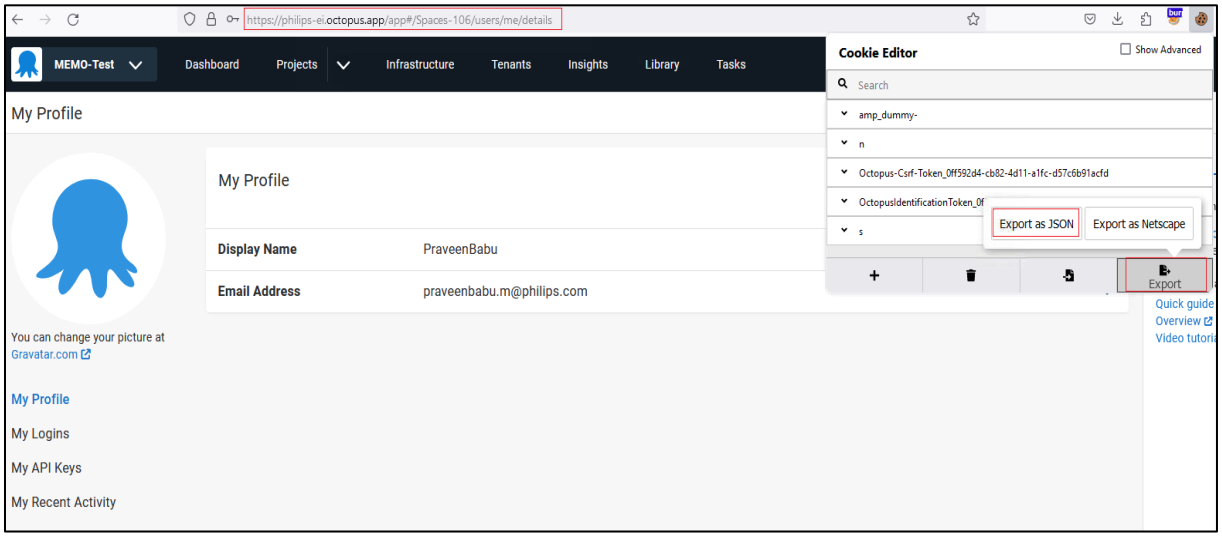
Step 1: In browser 1, log in to the application as user A.

Step 2: In browser 2, log in with same credentials of user A

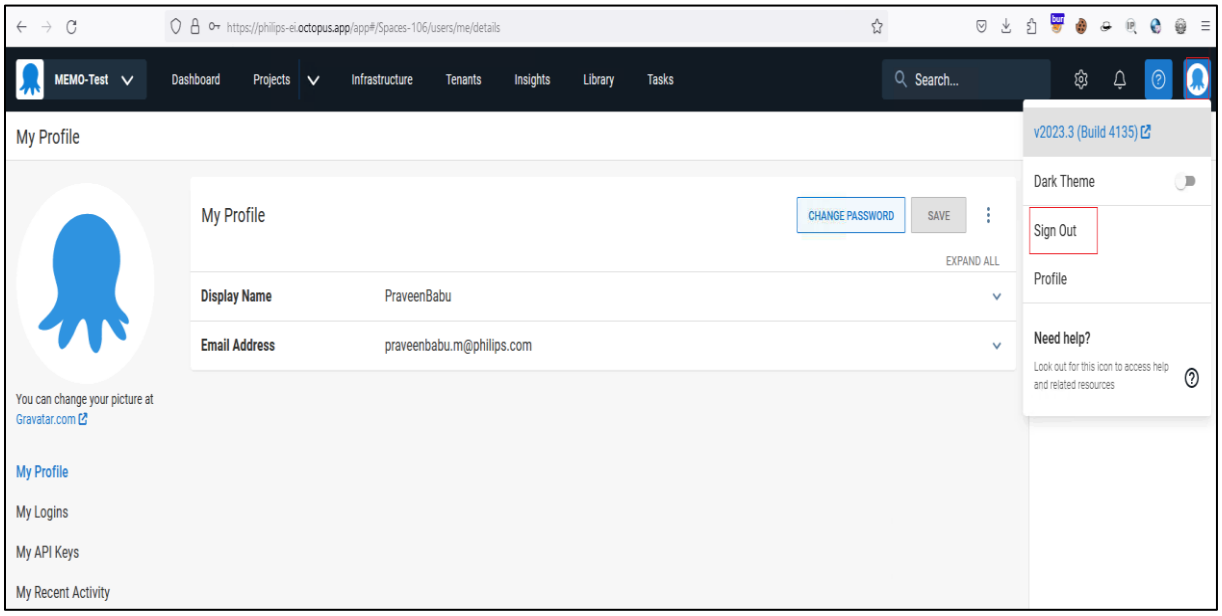
Step 3: The application not validates the concurrent logins.

Supportive Evidence:

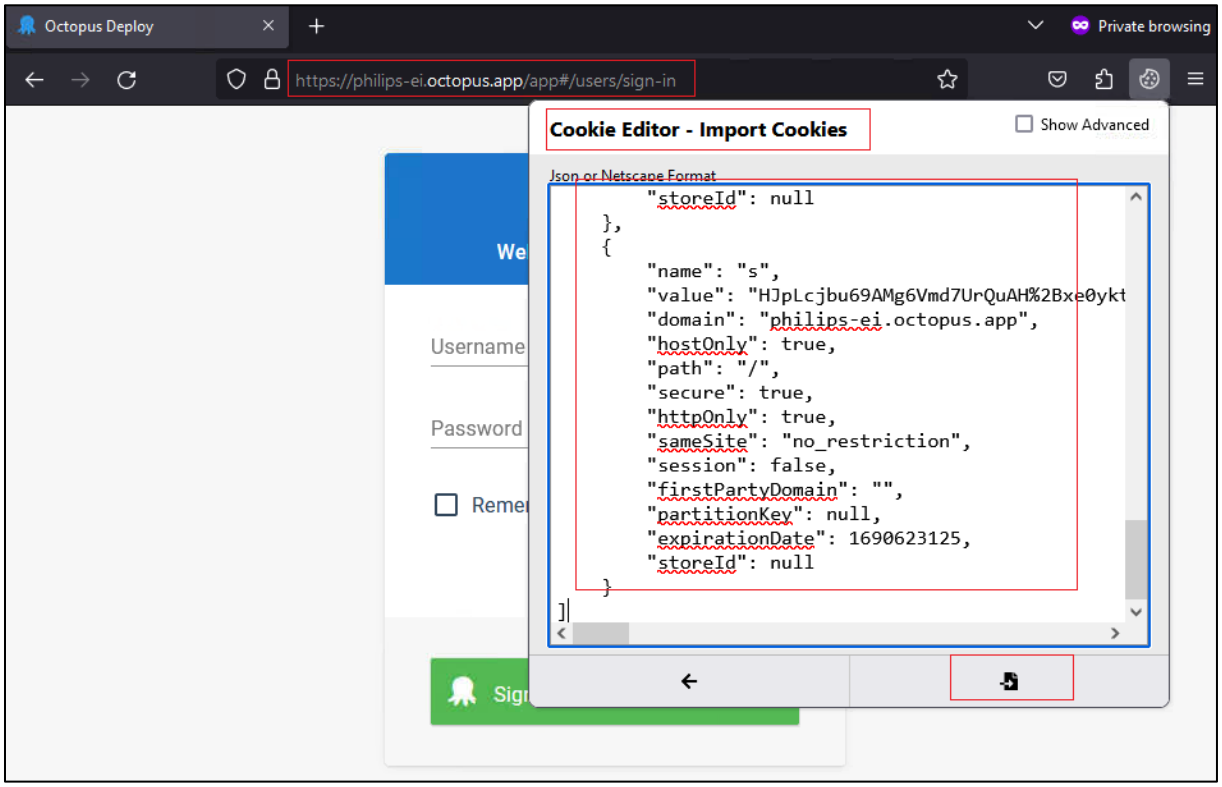
Case 1:



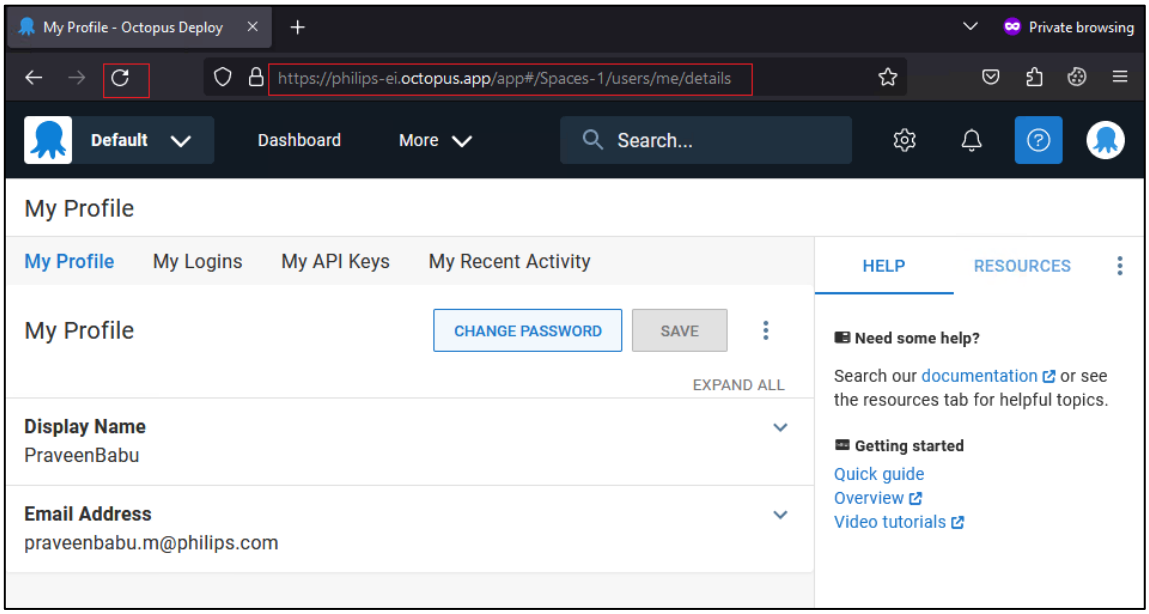
a. Login to application with valid credentials of user A in Browser 1 & export the cookies



b. Sign out of the user A in Browser 1

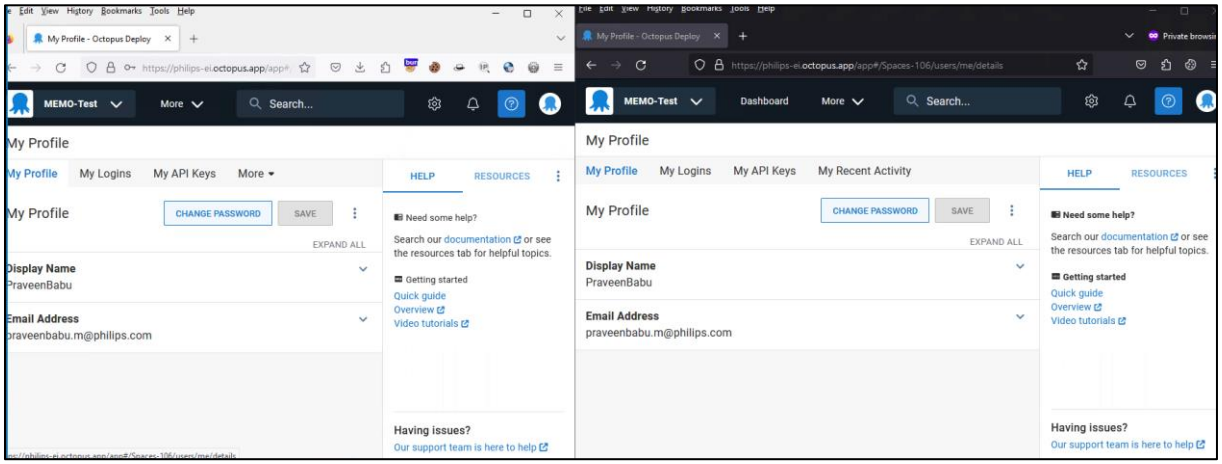


c. In Browser 2, open the application url, import the copied cookies



d. Once you clicked on refresh button in browser 2 after importing, application takes you to post login pages without valid credentials.

Case 2:



e. Application allows user to login in two different browser at same time, concurrent sessions allowed

8.12 Desktopapp: Improper Error Handling

Vulnerability Title	Improper Error Handling
Vulnerability Category	A5 Security Misconfiguration
Severity	**Low**
CVSS V3 Calculation	CVSS Base Score: 3.4 CVSS:3.1/AV:A/AC:H/PR:L/UI:R/S:U/C:L/I:L/A:N
Description	**Vulnerability Description** Improper handling of errors can introduce a variety of security problems for a web site. The most common problem is when detailed internal error messages such as stack traces, database dumps, and error codes are displayed to the user (attacker). These messages reveal implementation details which are supposed to be hidden. Reference: https://owasp.org/www-community/Improper_Error_Handling **Exploitability rational** An attacker should have access to the application. **Impact rational** By leveraging the verbose error an attacker can gain more information about the target which help in fine tuning his/her future attack.
Affected Binary	Octopus.Manager.Tentacle.exe
Recommendation	The application should return customized generic error messages to the user's browser. If details about the error are needed for debugging or support reasons a unique identifier may be created and displayed to the user along with the generic error message for reference. This same unique identifier can be included with the error that is logged to the server so that it can be easily correlated with the issue. References: • https://cheatsheetseries.owasp.org/cheatsheets/Error_Handling_Cheat_Sheet.html

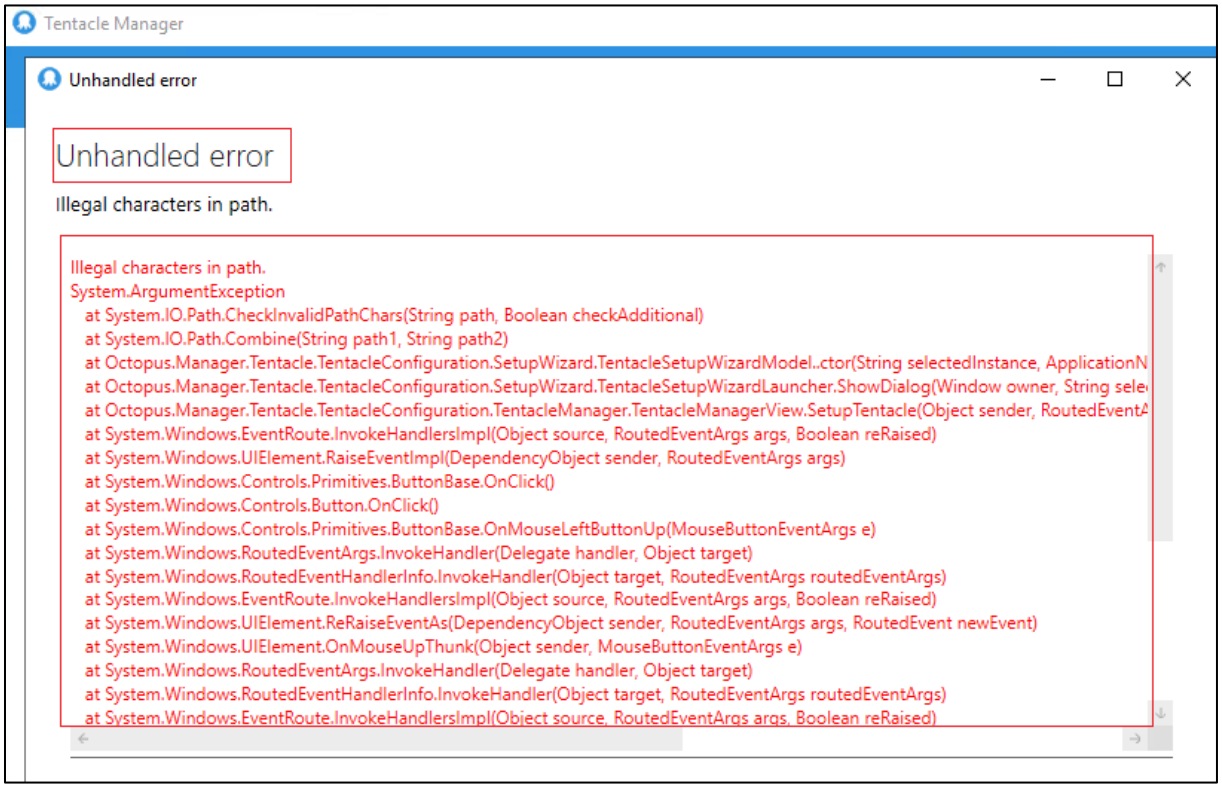
Status	Open

Test as of on 26Jul2023:

Steps to Reproduce:

Step 1: Open the Octopus Tentacle Manager.
Step 2: Click on Add tentacle.
Step 3: Add tentacle name as <script>alert(1)</script>.
Step 4: Click on Get started, the application responds back with unhandled error message.

Supportive Evidence:



8.13 API & DesktopApp & Webapp: Lack of Input Validation

Vulnerability Title	Lack of Input Validation
Vulnerability Category	A8- Security Misconfiguration
Severity	**Low**
CVSS V3 Calculation	CVSS Base Score: 2.6 CVSS:3.0/AV:N/AC:H/PR:L/UI:R/S:U/C:L/I:N/A:N
Description	**Vulnerability Description:** It was observed that the API accepts invalid characters as a input in the request. **Exploitability rational:** Without proper validation, quality and integrity issues may exist since data is sent to the application may cause a failure in business logic. Additionally, malformed content may corrupt server-side application processing that relies on properly formed user input to execute correctly. **Impact Rational:** An attacker could further exploit similar instance to perform attacks such as cross site scripting and injection related attacks.
Affected URLs & Binary	https://philipsinformatics-alertsink-test.eu1.phsdp.com/api/CRM/Case https://philipsinformatics-alertsink-dev.eu1.phsdp.com/api/CRM/Case Octopus.Manager.Tentacle.exe https://philips-ei.octopus.app/app#/
Recommendation	It is recommended that proper input validations should be performed at both server and client side.
Status	**OPEN**

Steps to Reproduce (API) :

- Visit the affected API endpoints in postman & Intercept with burpsuite
- Change the alertname paramter to any special chars or includes null values
- Observe that the same will be responded in API response.

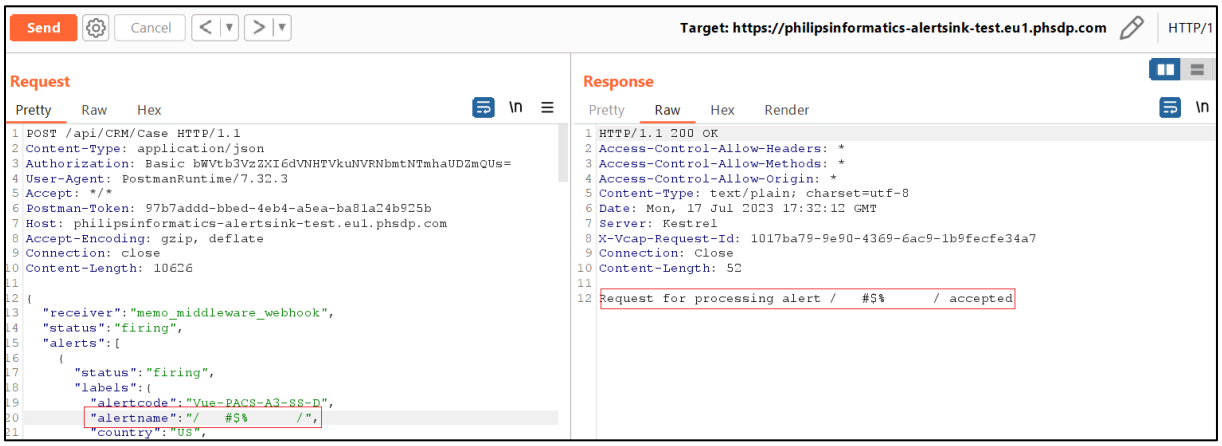
Steps to Reproduce (DesktopApp):

- Open the Octopus Tentacle Manager
- Click on Add tentacle
- Add tentacle name as <script>alert(1)</script>
- Click on Get started, the application responds back with unhandled error message.

Steps to Reproduce (WebApp):

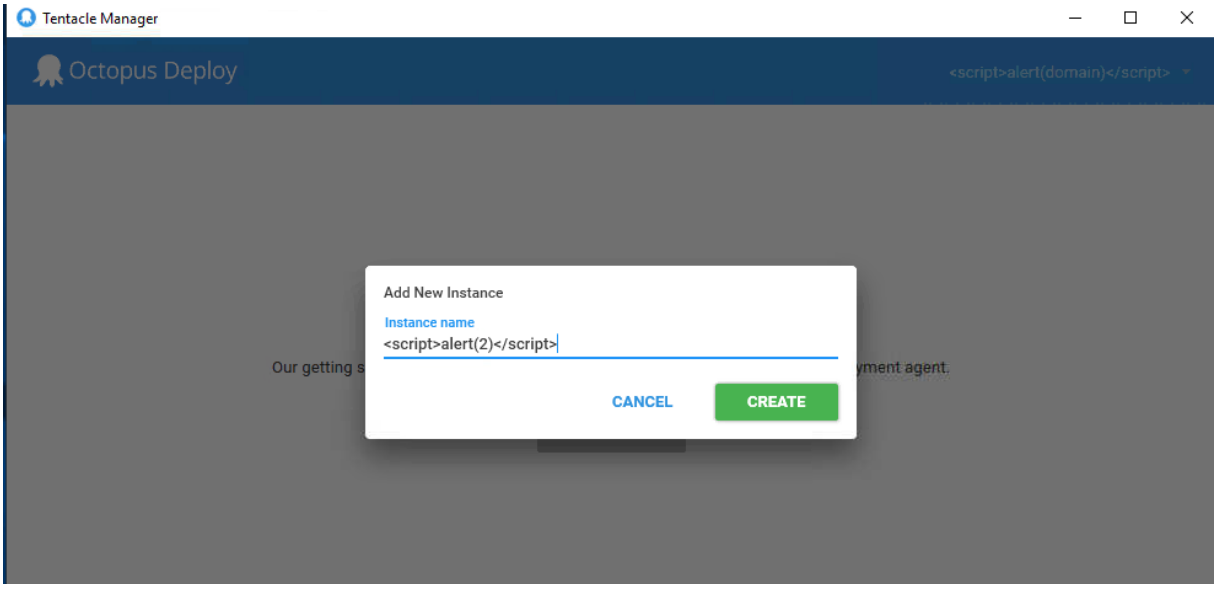
- Login to https://philips-ei.octopus.app/app#/ with valid credentials
- Navigate to Configuration section & Click on Users
- Add Username as <script>alert(1)</script>
- Application not sanitizing user input, it displays the same.

Supportive Evidence:

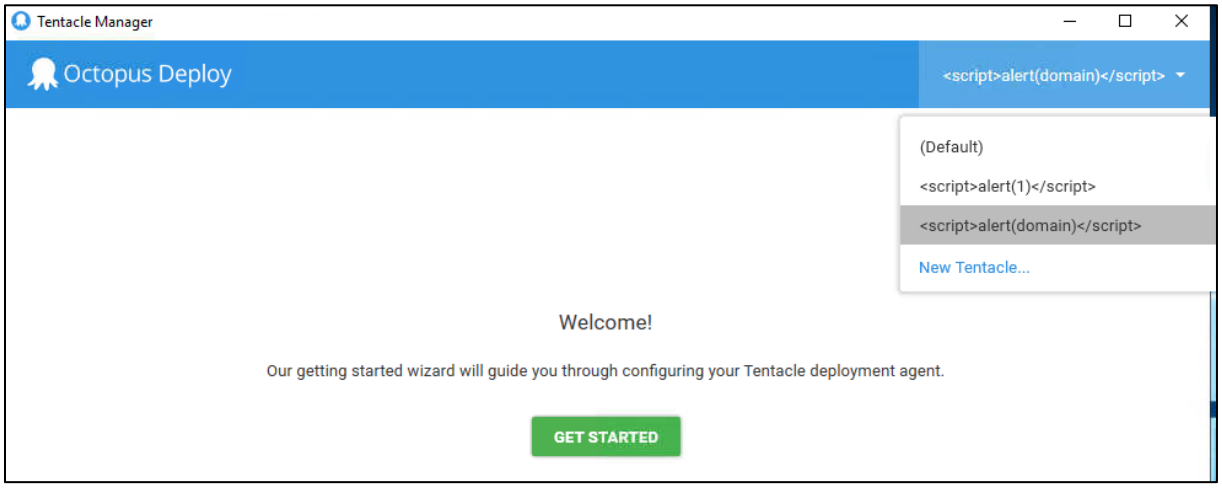


a. Screenshot shows that the CRM Case – Test – alertname value reflected back in response.

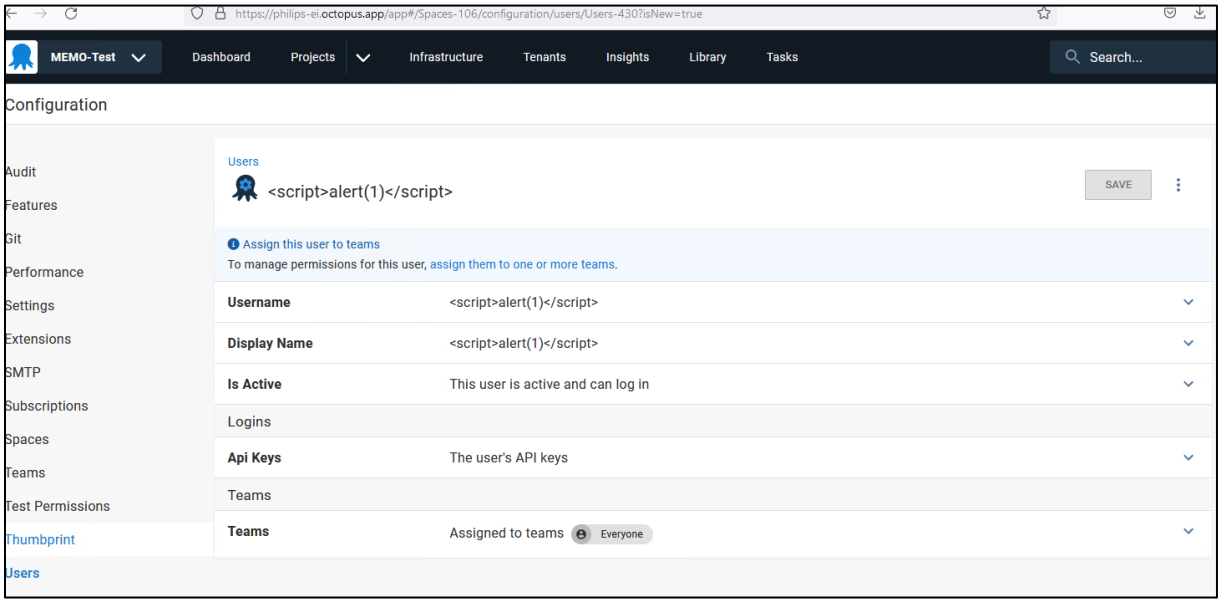
b. Screenshot shows that the CRM Case – dev – alertname value reflected back in response.



c. Screenshot shows that the Octopus Tentacle manager accepts the user input



d. Screenshot shows that the Octopus Tentacle manager accepts the user input



e. Screenshot shows that the Octopus application portal accepts the user input without validation

8.14 DesktopApp: Extraneous Services Enabled

Vulnerability Title	Extraneous Services Enabled
Vulnerability Category	A5- Security Misconfiguration
Severity	Low
CVSS V3 Calculation	CVSS Base Score: 3.7 CVSS:3.0/AV:N/AC:H/PR:N/UI:N/S:U/C:L/I:N/A:N
Description	**Vulnerability Description:** It was observed that application exposes more ports and services, which may be not necessary to be exposed in internal network. **Test as of on 15Jun2023:** Multiple High Risk Open Ports found. Should be closed if no business requirement. **Exploitability rational:** Unused services exposed to internal network appears to be not directly exploitable, until the attacker gets access to the network. **Impact Rational:** Exposed services that are not used or needed for the function of the server provide attackers with a great surface to launch attacks against the system and network. Unused services may be neglected and not updated, potentially introducing security vulnerabilities. The information available on the make of device could be used to attempt more sophisticated attacks against the infrastructure.
Affected Hosts	memoconc
Recommendation	It is recommended to review all the open ports and services and to disable or firewall services that are not in use to reduce the attack surface on the application.

Status	Open

Test as of on 15Jun2023:

Steps to Reproduce:

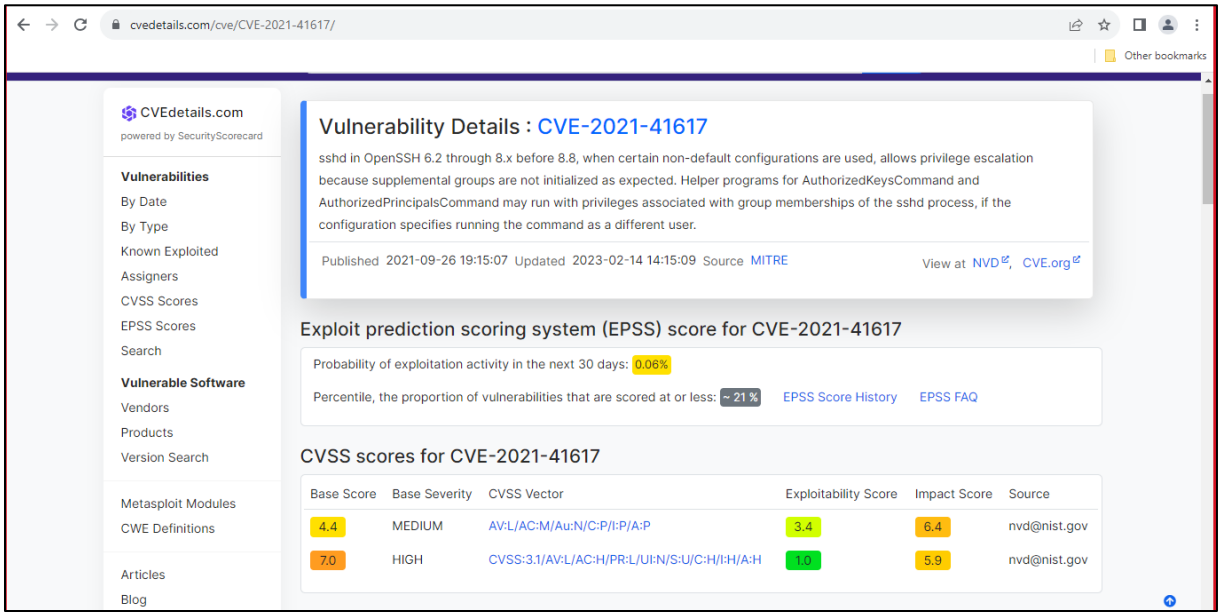
- Run nmap command line: nmap -sC -sV -Pn -p- <hosts>
- Observe the result that application exposing open ports & services

Supportive Evidence:

```
Nmap scan report for memoconc (130.147.140.173)
Host is up (0.00018s latency).
Other addresses for memoconc (not scanned): fe80::4186:505b:bf66:a0ab
Not shown: 983 closed tcp ports (reset)
PORT     STATE SERVICE       VERSION
22/tcp   open  ssh           OpenSSH 8.4 (protocol 2.0)
80/tcp   open  http          Microsoft IIS httpd 10.0
135/tcp  open  msrpc         Microsoft Windows RPC
139/tcp  open  netbios-ssn   Microsoft Windows netbios-ssn
445/tcp  open  microsoft-ds?
808/tcp  open  mc-nmf        .NET Message Framing
1040/tcp open  netsaint?
1433/tcp open  ms-sql-s      Microsoft SQL Server 2019 15.00.4073
1801/tcp open  msmq?
2103/tcp open  msrpc         Microsoft Windows RPC
2105/tcp open  msrpc         Microsoft Windows RPC
2107/tcp open  msrpc         Microsoft Windows RPC
3306/tcp open  mysql?
3389/tcp open  ms-wbt-server Microsoft Terminal Services
7778/tcp open  mc-nmf        .NET Message Framing
9000/tcp open  cslistener?
9090/tcp open  ssl/http      Golang net/http server (Go-IPFS json-rpc or InfluxDB API)
```

a. **Screenshot shows that the application exposes many open ports & services**

https://www.cvedetails.com/cve/CVE-2021-41617/



Identified CVE for the Openssh 8.4 application version.

8.15 Webservices: HTTPS Fallback Allowed

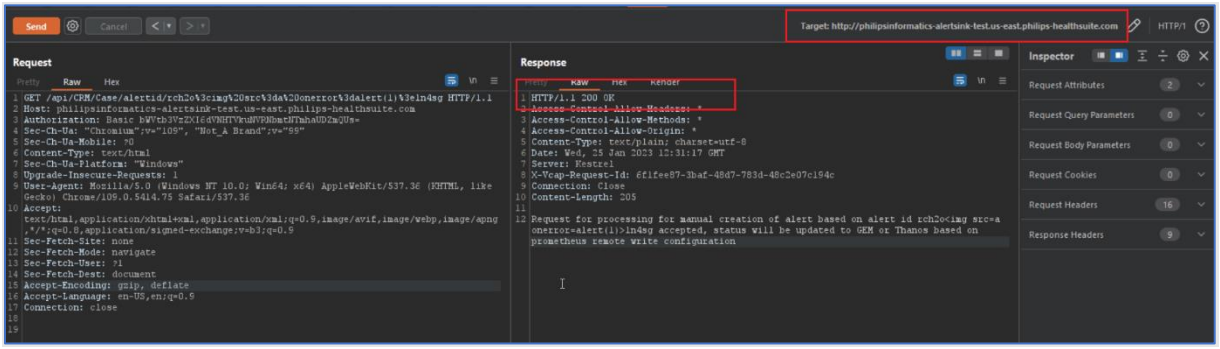
Vulnerability Title	HTTPS Fallback Allowed
Vulnerability Category	A5- Security Misconfiguration
Severity	**Low**
CVSS V3 Calculation	CVSS Base Score: 3.1 CVSS:3.1/AV:N/AC:H/PR:L/UI:N/S:U/C:L/I:N/A:N
Description	**Vulnerability Description:** During the security assessment of the Web services, it is found that the application may initially be served over HTTPS, it is also accessible over HTTP, resulting in application traffic that is sent in plaintext. Also, the API's are found missing HSTS header. **Retest as of on 15Jun2023:** 404 page not found. Hence considering this issue as closed. **Exploitability Rational:** If a victim accesses the API with an HTTP-based URL, an attacker listening on any network between the victim and the application server may view and modify the traffic. **Impact Rational:** Attacker can perform Man in the middle attack and can see all communication in clear text.
Affected Systems/IP Address/URL	https://philipsinformatics-alertsink-test.us-east.philips-healthsuite.com/
Recommendation	Enable HTTPS and enforce on the application server. Once HTTPS is properly configured, ensure that the application requires HTTPS for access to all application resources, including JavaScript files, style sheets, and images. When a user attempts to navigate to any part of the application over HTTP, the application should redirect the user to the HTTPS version of the application. Lastly, it is recommended that applications are deployed with HTTP Strict Transport Security (HSTS). HSTS forces the browser to access a site only over HTTPS and prevents access in cases where the authenticity of the X.509 certificate cannot be verified. HSTS is supported in recent versions of the Chrome, Firefox, and Opera web browsers.

	To enable HSTS, add the Strict-Transport-Security header to the response header when users first access the site over HTTPS. For more information on the HSTS header, refer to the relevant OWASP page, located at the below link: https://www.owasp.org/index.php/HTTP_Strict_Transport_Security
Status	**Closed**

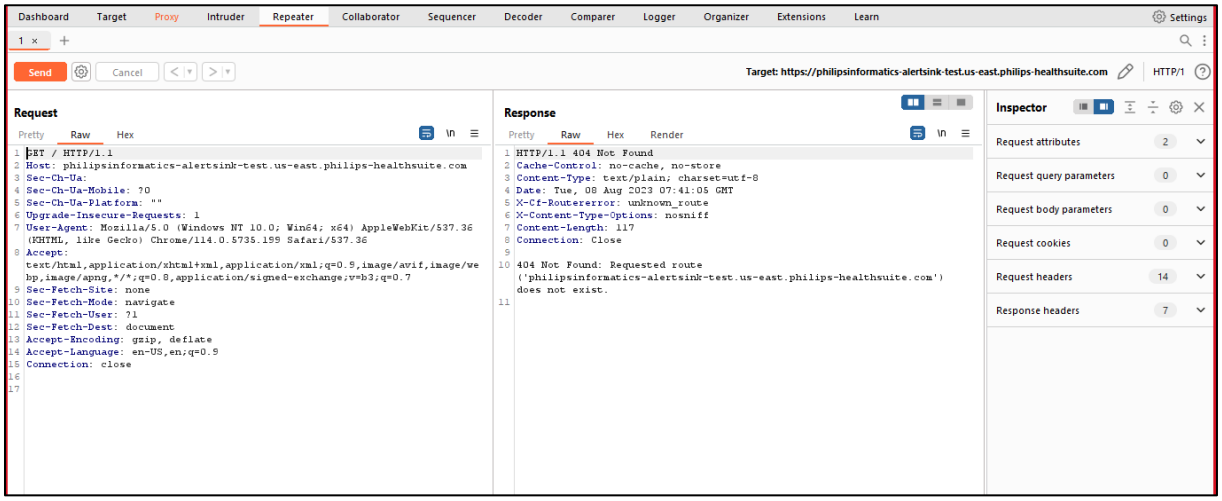
Steps to Reproduce:

- Send the create case request to burp repeater and edit the protocol from HTTPS to HTTP by changing the port number from 443 to 80.
- Send the request and you can observe that the backend supports HTTP.

Supportive Evidence:



Retest as of on 15Jun2023:



404 page not found. Hence issue has been closed.

8.16 WebApp & WebServices: Weak Input validation and Data Not Validated for Semantic Correctness

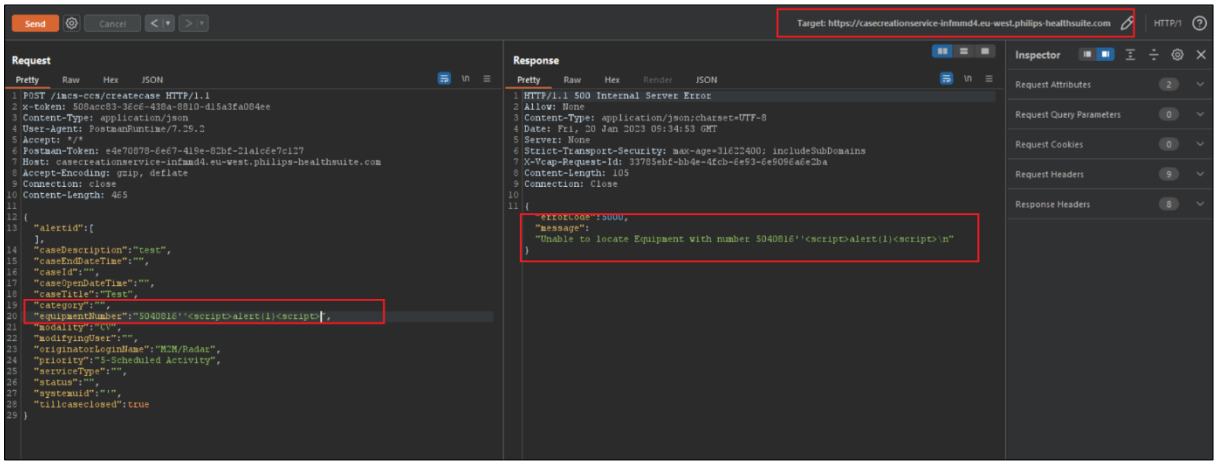
Vulnerability Title	Weak Input Validation and Data not Validated for Semantic Correctness
Vulnerability Category	A3- Injection
Severity	**Low**
CVSS V3 Calculation	CVSS Base Score: 3.5 CVSS:3.1/AV:N/AC:H/PR:L/UI:N/S:U/C:N/I:L/A:N
Description	**Vulnerability Description:** During the security assessment, it is observed that the application and the profile name change does not validate user input for semantic correctness. While there is server-side validation in place to ensure that user input does not negatively impact the application or other users, the application fails to ensure that the data supplied matches the expected format for each given field. **Retest as of on 15June2023:** Malicious Javascript Payload is not output encoded in response. Input Validation should be placed in the application. **Exploitability rational:** Without proper validation, quality and integrity issues may exist since data is sent to the application may cause a failure in business logic. Additionally, malformed content may corrupt server-side application processing that relies on properly formed user input to execute correctly. **Impact Rational:** • Damage to brand through site defacement. • Information leakage through injected page content that spoofs legitimate application functionality (For example, a form that asks the user to re-enter their credentials and then sends the credentials to the attacker).

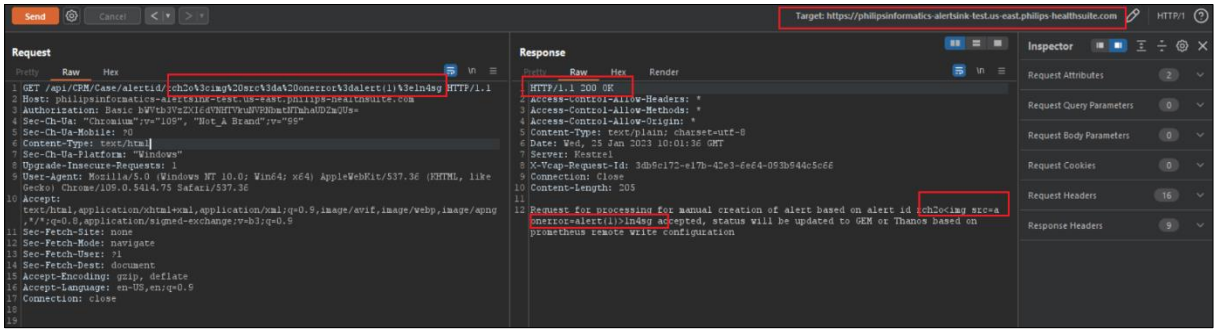
	• Page redirect to an attacker-controlled site via the meta refresh tag. Note that this attack is similar to cross-site scripting given that in both attack scenarios, injected payloads are rendered in a victim's browser. The difference lies in the capability each exploit provides the attacker, and the extent of the impact on the victim. Cross-site scripting allows an attacker to execute scripts in the victim's browser, whereas HTML injection only allows the attacker to inject malicious HTML into pages displayed to the victim.
Affected Systems/IP Address/URL	https://philipsinformatics-alertsink-test.us-east.philips-healthsuite.com/ https://casecreationservice-infmmd4.eu-west.philips-healthsuite.com/imcs-ccs/createcase **Retest as of on 15Jun2023:** https://mas-edi-grafana-client-test.eu-west.monitoring.hsdp.io/profile
Recommendation	Do a proper input validation on any data coming from client and always make sure uses output encoding while rendering any data on client side.
Status	**Open**

Steps to Reproduce:

- Launch postman tool and edit the post body of the create case API
- Edit the equipment number with malicious payload such as <script>alert(1)</script>
- You can observe that the payload is not output encoded in response.

Supportive Evidence:



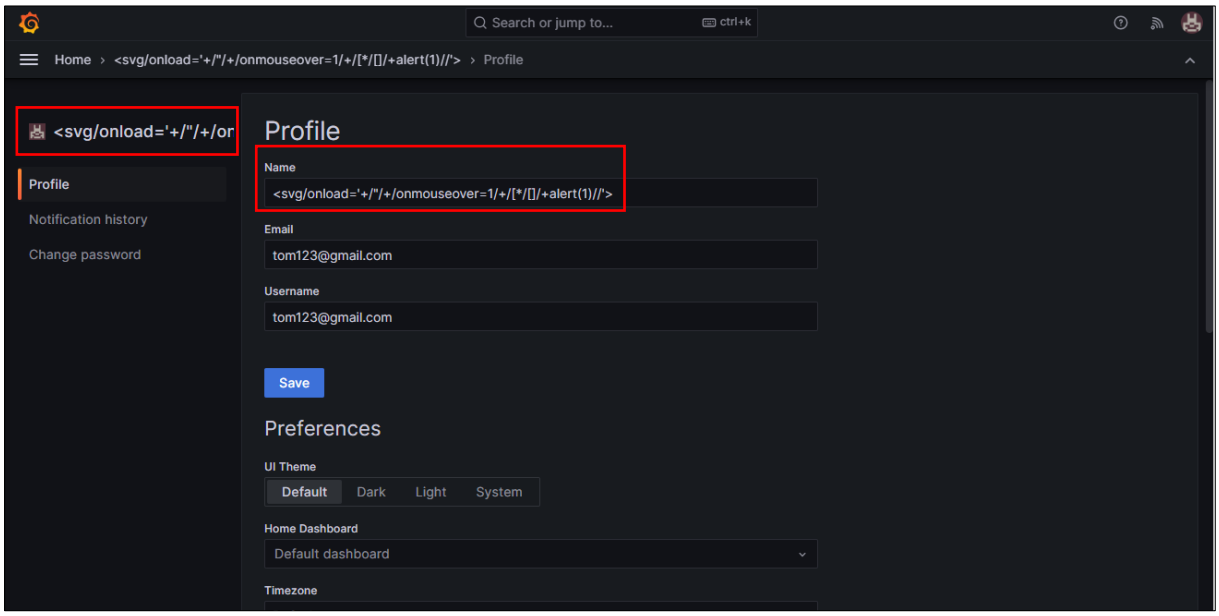


Retest results as of on 15June2023:

Steps to Reproduce:

- Open the application in the browser.
- Edit the profile name with malicious payload such as <svg/onload='+/"/+/onmouseover=1/+/[*/[]/+alert(1)//'>
- You can observe that the payload is not output encoded in response.

Supportive Evidence:



Payload is not output encoded in response.

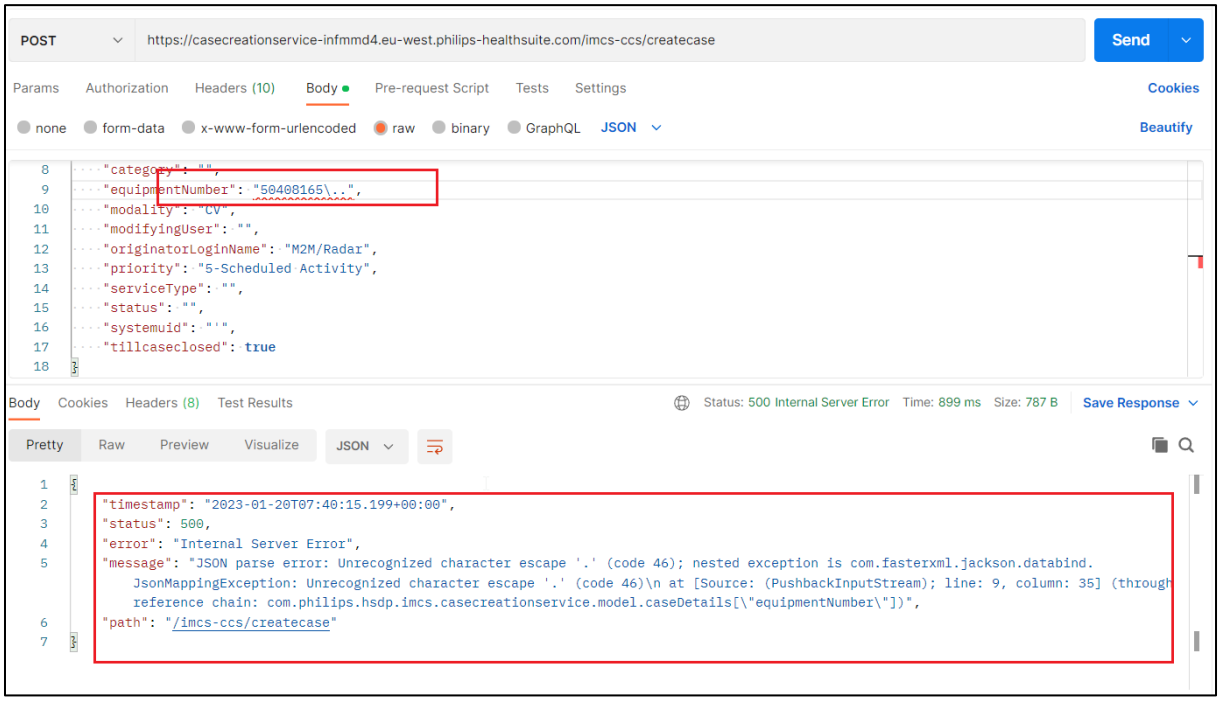
8.17 WebServices: Verbose Error Messages

Vulnerability Title	Verbose Error Messages
Vulnerability Category	A5-Security Misconfiguration
Severity	**Low**
CVSS V3 Calculation	CVSS Base Score: 3.1 CVSS:3.1/AV:N/AC:H/PR:L/UI:N/S:U/C:N/I:L/A:N
Description	**Vulnerability Description:** During the security assessment, it is observed that the backend provides a detailed error message in the form of a exceptional error when tried to input a malformed value that the server doesn't understand other than desired value in the input fields. **Exploitability rational** The attacker needs to be in internal network with a user privilege to perform enumeration. **Impact Rational:** In most cases, server information does not involve the leakage of critical pieces of information, but rather information that may aid the attacker through the exploitation phase of the attack.
Affected Systems/IP Address/URL	https://casecreationservice-infmmd4.eu-west.philips-healthsuite.com/imcs-ccs/createcase
Recommendation	It is always recommended to send custom error messages or no information on the error messages at all by providing only 400 Bad request response from the server if you get any unintended input.
Status	**Closed**

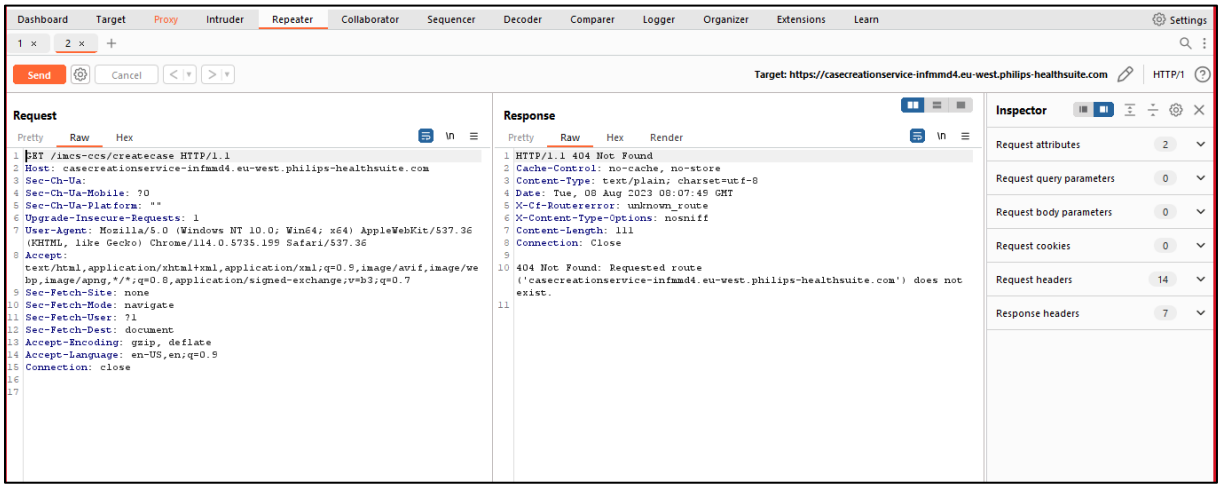
Steps to Reproduce:

- Edit the equipment number with some random data such as \.. and send the request.
- You can observe that the server responding with error messages.

Supportive Evidence:



Retest as of on 15Jun2023:



404 page not found. Hence considering this issue as closed.

9. Tools Used

Scope	Tools Used
Web Application Security and WebServices	Burpsuite, Postman
Desktop Application	HxD admin, Wireshark, EchoMirage, Sysinternals-ProcessExplorer, DotPeek, PE-SecurityPowershell

10. Automated Tool Report

NA

11. Manual Test Reports and Test Case Execution